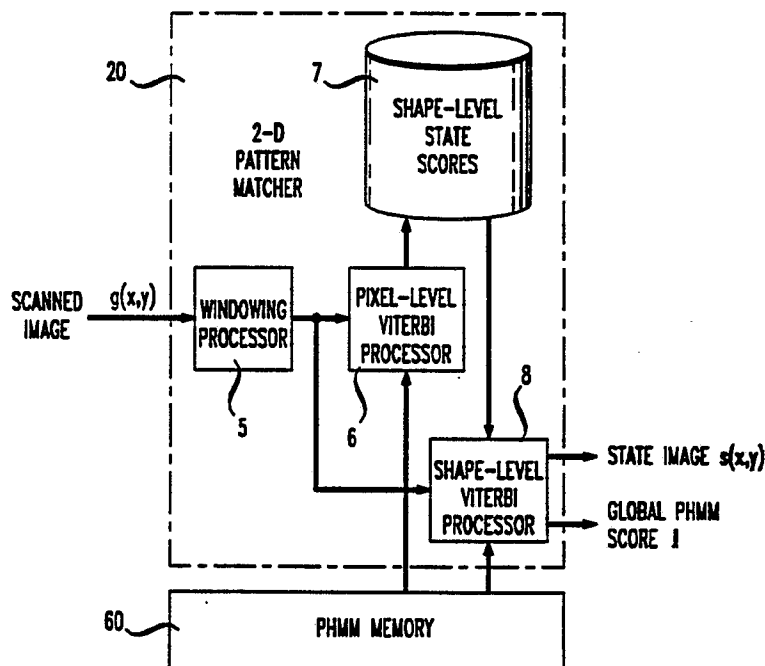


## INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification <sup>5</sup> :  G06K 9/52	A1	(11) International Publication Number: WO 93/18483 (43) International Publication Date: 16 September 1993 (16.09.93)
<p>(21) International Application Number: PCT/US93/01843</p> <p>(22) International Filing Date: 2 March 1993 (02.03.93)</p> <p>(30) Priority data: 07/844,810                      2 March 1992 (02.03.92)                      US</p> <p>ERREUR CODE DEPOSANT BFG: AMERICAN TELEPHONE AND TELEGRAPH COMPANY [US/US]; 32 Avenue of the Americas, New York, NY 10013-2412 (US).</p> <p>ERREUR CODE DEPOSANT BFG: LEVIN, Esther ; 15 Short Hills Circle, Millburn, NJ 07041 (US). PIERACCINI, Roberto ; 107 Midvale Avenue, Millington, NJ 07946 (US).</p>		<p>(74) Agents: WILDE, Peter, V., D. et al.; Post Office Box 679, Holmdel, NJ 07733 (US).</p> <p>(81) Designated States: CA, JP, US, European patent (DE, FR, GB, IT).</p> <p><b>Published</b> <i>With international search report. Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.</i></p>

## (54) Title: METHOD AND APPARATUS FOR IMAGE RECOGNITION



## (57) Abstract

A method for image recognition is provided which involves storing a plurality of two-dimensional hidden Markov models (60) each such model comprising a one-dimensional shape-level hidden Markov model comprising one or more shape-level states, each shape-level state comprising a one-dimensional pixel-level hidden Markov model comprising one or more pixel-level states. An image is scanned to produce one or more sequences of pixels. For a stored two-dimensional hidden Markov model, local Viterbi scores for a plurality of pixel-level hidden Markov models are determined for each sequence of pixels (6). A global Viterbi score of a shape-level hidden Markov model is determined based on a plurality of local Viterbi scores and the sequences of pixels. The scanned image is recognized based on one or more global Viterbi scores.

***FOR THE PURPOSES OF INFORMATION ONLY***

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	FR	France	MR	Mauritania
AU	Australia	GA	Gabon	MW	Malawi
BB	Barbados	GB	United Kingdom	NL	Netherlands
BE	Belgium	GN	Guinea	NO	Norway
BF	Burkina Faso	GR	Greece	NZ	New Zealand
BG	Bulgaria	HU	Hungary	PL	Poland
BJ	Benin	IE	Ireland	PT	Portugal
BR	Brazil	IT	Italy	RO	Romania
CA	Canada	JP	Japan	RU	Russian Federation
CF	Central African Republic	KP	Democratic People's Republic of Korea	SD	Sudan
CC	Congo	KR	Republic of Korea	SE	Sweden
CH	Switzerland	KZ	Kazakhstan	SK	Slovak Republic
CI	Côte d'Ivoire	LI	Liechtenstein	SN	Senegal
CM	Cameroon	LK	Sri Lanka	SU	Soviet Union
CS	Czechoslovakia	LU	Luxembourg	TD	Chad
CZ	Czech Republic	MC	Monaco	TG	Togo
DE	Germany	MG	Madagascar	UA	Ukraine
DK	Denmark	ML	Mali	US	United States of America
ES	Spain	MN	Mongolia	VN	Viet Nam
FI	Finland				

## METHOD AND APPARATUS FOR IMAGE RECOGNITION

### Field of the Invention

The present invention relates generally to the field of image recognition, and specifically to pattern based image recognition.

### 5 Background of the Invention

Signal recognition systems operate to label, classify, or otherwise recognize an *unknown* signal. Signal recognition may be performed by comparing characteristics or features of unknown signals to those of *known* signals.

Features or characteristics of known signals are determined by a process  
10 known as *training*. Through training, one or more samples of known signals are examined and their features or characteristics recorded as *reference patterns* in a database of a signal recognizer.

To recognize an unknown signal, a signal recognizer extracts features from the signal to characterize it. The features of the unknown signal are referred to  
15 as the *test pattern*. The recognizer then compares each reference pattern in the database to the test pattern of the unknown signal. A scoring technique is used to provide a relative measure of how well each reference pattern matches the test pattern. The unknown signal is recognized as the reference pattern which most closely matches the unknown signal.

20 There are many types of signal recognizers, *e.g.*, *template*-based recognizers and *hidden Markov model (HMM)* recognizers. Template-based recognizers are trained using first-order statistics based on known signal samples (*e.g.*, spectral means of such samples) to build reference patterns. Typically, scoring is accomplished with a time registration technique, such as *dynamic time warping*  
25 (*DTW*). *DTW* provides an optimal time alignment between reference and test patterns by locally shrinking or expanding the time axis of one pattern until that pattern optimally matches the other. *DTW* scoring reflects an overall *distance* between two optimally aligned reference and test patterns. The reference pattern having the lowest score (*i.e.*, the shortest distance between itself and the test pattern)  
30 identifies the test pattern.

HMM recognizers are trained using both first and second order statistics (*i.e.*, means and variances) of known signal samples to build reference patterns. Each reference pattern is an N-state statistical model incorporating these means and variances. An HMM is characterized by a state transition matrix, A (which provides  
35 a statistical description of how new states may be reached from old states), and an

observation probability matrix, B (which provides a description of which spectral features are likely to be observed at a given state). Scoring of a test pattern reflects the probability of the sequence of features in the pattern given a model (*i.e.*, given a reference pattern). Scoring across all models may be provided by conventional  
5 dynamic programming techniques, such as Viterbi scoring well known in the art. The HMM which indicates the highest probability of the sequence of features in the test pattern identifies the test pattern.

Pattern-based signal recognition techniques, such as DTW and HMMs, have been applied in the past to the one-dimensional problem of speech recognition,  
10 where unknown signals to be recognized are speech signals and the one dimension is time. It has been a problem of some interest to provide for multi- dimensional signals, such as two-dimensional image signals, a set of general tools analogous to those available for one-dimensional signal recognition.

### Summary of the Invention

15 The present invention provides a method and apparatus for multi-dimensional signal recognition. The invention accomplishes recognition through multi-dimensional reference pattern scoring techniques.

An illustrative embodiment of the present invention provides a two-dimensional image recognizer for optical character recognition. The recognizer is  
20 based on *planar hidden Markov models (PHMMs)* with constrained transition probabilities. Each PHMM comprises a one-dimensional *shape-level* hidden Markov model and represents a single image reference pattern. A shape-level HMM comprises one or more *pixel-level* hidden Markov models, each of which represents a localized portion of a shape-level HMM. The embodiment operates to determine,  
25 for a given PHMM and a given sequence of pixels in an unknown character image, a local Viterbi score for each of one or more pixel-level HMMs in a shape-level HMM. Furthermore, the embodiment operates to determine a global Viterbi score for a shape-level HMM based on the plurality of local Viterbi scores. Character images are recognized based on the global Viterbi scores. A global Viterbi score is  
30 provided for each PHMM (*i.e.*, each shape-level HMM) reference pattern.

### Brief Description of the Drawings

Figure 1 presents illustrative groupings of pixel-level hidden Markov model states.

Figure 2 presents the illustrative groupings of pixel-level hidden Markov model states from Figure 1 associated with the shape-level states of a shape-level hidden Markov model.

Figure 3 presents a shape-level hidden Markov model comprising the  
5 shape-level states presented in Figure 2.

Figure 4 presents an illustrative optical character recognition system according to the present invention.

Figure 5 presents components of the two-dimensional pattern matcher presented in Figure 4.

10 Figure 6 presents an image of a scanned character, T, comprising a plurality of linear pixel sequences.

### Detailed Description

#### Introduction

An illustrative optical character recognition system according to the  
15 present invention includes a plurality of two-dimensional (or planar) hidden Markov models to represent images to be recognized. Each planar hidden Markov model is *defined* by:

i. a set of pixel-level states:

$$S = \{ s(x,y) \}, x=1, \dots, X, y=1, \dots, Y;$$

20 ii. a set of transition probabilities:

$$A_{(i,j),(k,l),(m,n)} \equiv P(s(x,y) = (m,n) \mid s(x-1,y) = (i,j), s(x,y-1) = (k,l)), (1)$$

where x and y are abscissa and ordinate, respectively, in a conventional two-dimensional coordinate system; and

iii. a set of observation probability densities  $B(x,y)$ , one for each state  $s(x,y)$ .

25 Each two-dimensional hidden Markov model may be *represented* as a set of *shape-level states*  $G_1, G_2, \dots, G_{N_G}$ . Each shape-level state,  $G_j$ , corresponds to a particular *grouping* of one or more *pixel-level states*,  $S$ . According to the principles expressed in the *Appendix hereto*, these groupings of pixel-level states should satisfy the following conditions:

30 a. The number of groups of shape-level states,  $N_G$ , is a polynomial function of the number of pixel-level states  $X \times Y$ .

- b. The union of all groups of shape-level states,  $G = \bigcup G_j$ , coincides with the set of pixel-level states,  $S$ .

With respect to the groups of shape-level states, the transition probabilities should fulfill the two following conditions:

- 5 c.  $A_{(i,j),(k,l),(m,n)} \neq 0$  only if there exists  $p$ ,  $1 \leq p \leq N_G$ , such that  $(i,j), (m,n) \in G_p$ ; and (2)

- d.  $A_{(i,j),(k,l),(m,n)} = A_{(i,j),(k_1,l_1),(m,n)}$  if there exists  $p$ ,  $1 \leq p, r \leq N_G$ , such that  $(k,l), (k_1,l_1) \in \gamma_p$ , (3)

$$\frac{A_{(i,j),(k,l),(m,n)}}{A_{(i,j),(k_1,l_1),(m,n)}} = \frac{A_{(i,j),(k,l),(m_1,n_1)}}{A_{(i,j),(k_1,l_1),(m_1,n_1)}} = \frac{A_{(i_1,j_1),(k,l),(m,n)}}{A_{(i_1,j_1),(k_1,l_1),(m,n)}}$$

- 10 if there exists  $p$ ,  $1 \leq p \leq N_G$ , such that  $(i,j), (i_1,j_1), (m,n), (m_1,n_1) \in \gamma_p$ , and where  $(k,l) \in \gamma_r, (k_1,l_1) \in \gamma_r$ .

An example of the application of these conditions (a–d) is presented in Figures 1 - 3. In Figure 1, seven shape-level states,  $G_1$  to  $G_7$ , are shown with reference to a  $4 \times 4$  matrix of pixel-level states. As shown in Figure 2, each shape-level state,  $G_j$ , corresponds to a one-dimensional pixel-level hidden Markov model comprising four pixel-level states. Moreover, each shape-level state,  $G_j$ , is but one state in a shape-level hidden Markov model, as shown in Figure 3. The arrows between states in the HMM of Figure 3 indicate *legal* state transitions within the constraints of conditions c and d, *above*.

- 20 The transition probabilities among the pixel-level and shape-level states are derived from  $A_{(i,j),(k,l),(m,n)}$ . When conditions c and d hold for a particular grouping, then transition probability  $A_{(i,j),(k,l),(m,n)}$  can be represented as:

$$A_{(i,j),(k,l),(m,n)} = A_{(i,j),(m,n)}^p \times \alpha_{rp}, \quad (4)$$

where

$$A_{(i,j),(m,n)}^p = P(s(x,y)=(m,n) \mid (s(x-1,y)=(i,j), (m,n), (i,j) \in G_p)), \quad (5)$$

and

$$\alpha_{rp} = P(s(x,y) \in G_p \mid s(x,y-1) \in G_r). \quad (6)$$

Hence, (5) defines the transition probabilities between pixel-level states in a one-dimensional pixel-level HMM (such as, *e.g.*, any of those appearing in Figure 2, and (6) defines the transition probabilities between shape-level states in a one-dimensional shape-level HMM. By virtue of (i) the nesting of pixel-level  
5 HMMs in a shape-level HMM, and (ii) conditions c and d specified above, a general two-dimensional (or planar) HMM for use in image recognition is provided. Note that the pixel-level state observation probabilities are *not* affected by the grouping of states.

### An Illustrative Embodiment

10 For clarity of explanation, the illustrative embodiment of the present invention is presented as comprising individual functional blocks (including functional blocks labeled as "processors"). The functions these blocks represent may be provided through the use of either shared or dedicated hardware, including, but not limited to, hardware capable of executing software. (Use of the term "processor"  
15 should not be construed to refer exclusively to hardware capable of executing software.) Illustrative embodiments may comprise digital signal processor (DSP) hardware, such as the AT&T DSP16 or DSP32C, and software performing the operations discussed below. Very large scale integration (VLSI) hardware embodiments of the present invention, as well as hybrid DSP/VLSI embodiments,  
20 may also be provided.

Figure 4 presents an illustrative optical character recognition system according to the present invention. The system comprises a conventional image scanner 10, a two-dimensional pattern matcher 20, control switches R and T, a decision processor 30, a state image memory 35, a probability estimation processor  
25 45, and a planar hidden Markov model memory 40.

The conventional image scanner 10 receives a physical image of a character and scans it to generate as output a matrix signal,  $g(x,y)$ . This signal represents the intensity of the physical image at each pixel location,  $x,y$ , within the image. PHMMs, developed through a training process discussed below, are stored in  
30 the PHMM memory 40. Each PHMM in memory 40 represents a character to be recognized in an optical character application.

The matrix signal for the image,  $g(x,y)$ , is processed by the two-dimensional pattern matcher 20 to generate, for each PHMM, a global Viterbi score 1 resulting from the comparison of the PHMM and the signal  $g(x,y)$ . A state image,  
35  $s(x,y)$ , is also generated to represent the index of the PHMM state corresponding to

pixel,  $x, y$ .

The two-dimensional pattern matcher 20 is presented in Figure 5. Pattern matcher 20 comprises a windowing processor 5, a pixel-level Viterbi processor 6, a local-level score memory 7, and a shape-level Viterbi processor 8.

5 The windowing processor 5 receives the matrix signal,  $g(x, y)$ , and extracts therefrom successive sequences of pixels,  $L_1, L_2, \dots, L_M$ . As shown illustratively in the example of Figure 6, these sequences may be linear sequences of pixels.

The pixel-level Viterbi processor 6 determines for each pixel sequence  
10  $L_i$  and each group  $G_j$  (comprising a pixel-level HMM) a local state score  $d_{ij}$ . This is done by computing the Viterbi score of the linear sequence of pixels,  $L_i$ , with the pixel-level linear HMM,  $G_j$ . An  $N_G \times M$  matrix of the local-level state scores is stored in memory 7.

The shape-level Viterbi processor 8 computes a global score for a given  
15 PHMM as the Viterbi score of a linear shape-level hidden Markov model using the sequence  $L_i$  as the observation sequence and  $d_{ij}$  as the local state score for each shape-level state  $G_j$  and each observation  $L_i$ . Also, the state image,  $s(x, y)$ , is computed using conventional backtracking methods for hidden Markov models.

The operations performed by the two-dimensional pattern matcher 20  
20 are repeated for each PHMM in the PHMM memory 40. In recognition mode (*i.e.*, when switch R is closed and switch T is open), the decision processor 30 recognizes the scanned image as the character corresponding to the PHMM with the highest score,  $I_h$ .

In training mode, switch T is closed and switch R is open. The training  
25 mode operation of the embodiment involves conventional Viterbi training of a linear hidden Markov model. Known samples of all characters to be recognized are provided sequentially as input to scanner 10. For each such sample of a given character, a state image  $s(x, y)$  is determined by the two-dimensional pattern matcher 20 as described above, using only the PHMM corresponding to the known  
30 sample. All known samples for the character are processed in this fashion, with each state image  $s(x, y)$  stored in state image memory 35. Once all such samples for a character are processed and the resulting state images are stored, the probability estimation processor 45 estimates new transition and observation probabilities for the PHMM (as frequency counts) in conventional fashion taking into account the  
35 conditions c and d described above for the state transition probabilities.



## APPENDIX

## 1. Introduction

In this appendix we extend the dynamic time warping (DTW) algorithm, widely used in automatic speech recognition (ASR), to a dynamic plane warping (DPW) algorithm, for applications in the field of optical character recognition (OCR) or similar applications.

This appendix is written from the point of view of a "speech-researcher"; i.e., we start with the description of the single-dimensional case and then extend it to two dimensions in order to point out the similarity and the differences between the two algorithms. No previous knowledge about speech recognition is assumed.

In the next section we first discuss the general template matching approach to pattern recognition and show the role of DTW or DPW algorithms in this paradigm. Then we describe the single-dimensional warping, or time alignment problem, and show how the DTW algorithm solves the problem for template-based systems in polynomial time using a general principle of optimality. The two-dimensional warping problem is defined in section 2.2, and its general solution using the same optimality principle is presented. Although the application of the optimality principle in this case reduces the computational complexity of planar warping, the complexity still remains exponential in the dimensions of the image. We show that by restricting the original warping problem, by limiting the class of possible distortions somewhat, we can reduce the computational complexity dramatically, and find the optimal solution to the restricted problem in polynomial time. This approach differs from the one taken in references [1] and [2], where instead of restricting the problem, a suboptimal solution to the general problem was found. In section 3, the statistical modeling approach to pattern recognition is described. In section 3.1, we discuss statistical modeling of temporal

signals using HMM, and show how this approach is more general, but still similar to DTW. In section 3.2, we introduce the planar hidden Markov model (PHMM) that, on one hand, extends the HMM concept to model images and, on the other hand, generalizes the DPW approach. We show that the restricted formulation of the planar warping problem of section 2.2.3 is equivalent to zeroing some transition probabilities in the PHMM. In section 4, experimental results of isolated hand-written digit recognition experiments are presented. The results indicate that even in the simple case of isolated characters, the elimination of planar distortions enhances the performance significantly. We anticipate that the advantage of this approach will be even more prominent in harder tasks, such as cursive writing recognition/spotting, that involve some of the above mentioned problems. The major ideas of this appendix are summarized in section 5.

## 2. Template Matching Approach to Pattern Recognition

The task of pattern recognition is that of classifying a set of measured patterns ( e.g., acoustic signals, pixel map images, etc.) into a finite set  $C = \{C_1, \dots, C_{N_c}\}$  of distinct classes representing spoken words or phonemes in the case of speech recognition, and written words or characters in the OCR task. Template matching is one of the many possible ways to solve this problem. According to this approach, each class is represented by a template (a reference pattern), and a new pattern is classified by selecting the class  $C_k$  for which the distance  $D_k$  between the new pattern and the class representative template is minimal, i.e.

$$k = \arg \min_{1 \leq n \leq N_c} D_n. \quad (1)$$

The difficulty in the pattern recognition task arises because of the intra-class variability of the patterns. Methods have to be developed to reduce such variability, thereby building up some invariance properties for the classifier. This intra-class variability is sometimes caused by non-linear distortions during the generation process of the patterns. In speech recognition this problem is known as the 'time alignment' problem, and its source is the temporal variability of the spoken utterances. The DTW procedure described below attempts to reduce the magnitude of this problem. The purpose of the procedure is to time-align the test and the reference patterns by stretching and contracting the test pattern to optimally match it to the reference, by minimizing a measure of the spectral distance  $D_k$  between the time-aligned patterns. temporal distortions.

The problem of intra-class variability also arises in optical character recognition due to non-linear, non-uniform elastic distortions (i.e., stretching, contracting) of the hand-written characters. In this appendix we show how to address this problem by generalizing the DTW procedure for planar alignment of images.

## 2.1 Matching Temporal Signals

**2.1.1 One-Dimensional Problem Formulation** The DTW algorithm is a procedure that was developed for optimally aligning two temporal signals:  $G_R^k = \{g_R^k(t) : 1 \leq t \in \mathbf{Z}^+ \leq T_R, g_R^k(\cdot) \in G \subset \mathbf{R}^n\}$ , the reference or template signal, representing the  $k$ -th class, and  $G = \{g(t) : 1 \leq t \in \mathbf{Z}^+ \leq T, g(\cdot) \in G \subset \mathbf{R}^n\}$ , the test signal to

-10-

be classified.  $Z^+$  is the set of positive integers, and  $R^n$  is the  $n$ -dimensional real space. The goal of DTW is to find a mapping function  $\tilde{t}=f(t)$  that maps the test time scale to the reference time scale, such that the distortion

$$D_k = D(G_R^k, G) = \sum_{\substack{t=1 \\ \tilde{t}=f(t)}}^{T_R} d(g_R^k(\tilde{t}), g(t)) \quad (2)$$

between the aligned patterns is minimal, where  $d(\cdot, \cdot)$  is a defined local distance measure in  $G$ . For simplicity of notation we omit the class index  $k$  hereafter. The mapping function is constrained by global constraints, such as the boundary conditions,

$$f(1)=1, f(T)=T_R, \quad (3)$$

where we assume that the beginnings and the ends of the two patterns line up, and local monotonicity constraints, such as,

$$\Delta f = f(t+1) - f(t) \geq 0, \quad (4)$$

that prevent the mapping from "folding backwards" in time. We denote by  $f$  the set of all mapping functions that satisfy (3) and (4). Constraints (3) and (4) are typical, but not unique. Since the treatment of other kinds of global and local constraints is similar, we continue with the problem defined by (3) and (4) only.

**2.1.2 The Procedure** The problem of finding the optimal mapping has an exponential complexity since there are  $O(T_R^T)$  possible mappings in  $f$ . These mappings are shown as a set of paths in a time-time grid (Fig.1), where each path is a monotonically

increasing curve that starts at point  $A=(t=1, \tilde{t}=1)$  and ends at point  $B=(t=T, \tilde{t}=T_R)$ . The DTW algorithm finds the optimal alignment curve among all possible paths in polynomial time, using the dynamical programming optimality principle.<sup>[3]</sup> The optimality principle is based on the fact that the optimal alignment curve (i.e., the one with the minimal distortion along the path) connecting point  $A$  to point  $B$  through point  $C$  is found among all curves that optimally connect  $A$  and  $C$ . This basic principle leads to an efficient iterative procedure for finding the optimal curve connecting  $A$  and  $B$ .

In the  $n$ -th step of the procedure,  $2 \leq n \leq T$ , we assume that the optimal warping of the  $(n-1)$ -th interval of the test signal  $g(t)$ ,  $1 \leq t \leq n-1$ , to the  $i$ -th interval of the reference signal  $g_R(\tilde{t})$ ,  $1 \leq \tilde{t} \leq i$ , is known for all  $1 \leq i \leq T_R$ . Each optimal warping is defined by a mapping  $f_{i,n-1}$  and a distortion  $D_{i,n-1}$ , such that

$$D_{i,n-1} = \min_{f \in \mathbf{f}_{i,n-1}} \sum_{t=1}^{n-1} d(g_R(\tilde{t}), g(t)) ; \quad (5)$$

$$\tilde{t}=f(t)$$

$$f_{i,n-1} = \arg \min_{f \in \mathbf{f}_{i,n-1}} \sum_{t=1}^{n-1} d(g_R(\tilde{t}), g(t)) .$$

$$\tilde{t}=f(t)$$

Here we denote by  $\mathbf{f}_{i,n}$ ,  $1 \leq i \leq T_R$ ,  $1 \leq n \leq T$  the set of all mapping functions from an interval  $1 \leq t \leq n$  to an interval  $1 \leq \tilde{t} \leq i$ , satisfying the local monotonicity conditions (4) on their domain and global boundary conditions:

$$\text{for all } f \in \mathbf{f}_{i,n} \quad 1=f(1) ; i=f(n) . \quad (6)$$

It is clear that  $\mathbf{f}_{T_R,T}=\mathbf{f}$ . The warping  $f_{i,n-1}$  corresponds to a curve in the time-time grid

-12-

that optimally connects point A to the point  $(t=n-1, \tilde{t}=i)$ .

At this stage we can find the optimal warping of the  $n$ -th interval of the test signal to the  $i$ -th interval of the reference signal, namely  $f_{i,n}$  and  $D_{i,n}$ , for all  $1 \leq i \leq T_R$ .

$$\begin{aligned}
 D_{i,n} &= \min_{f \in \mathcal{F}_{i,n}} \sum_{t=1}^n d(g_R(\tilde{t}), g(t)) = & (7a) \\
 & \quad \tilde{t}=f(t) \\
 &= \min_{1 \leq j \leq i} \min_{f \in \mathcal{F}_{j,n-1}} \sum_{t=1}^{n-1} d(g_R(\tilde{t}), g(t)) + d(g_R(\tilde{t}=i), g(t=n)) = \\
 & \quad \tilde{t}=f(t) \\
 &= \min_{1 \leq j \leq i} D_{j,n-1} + d(g_R(\tilde{t}=i), g(t=n)) ,
 \end{aligned}$$

and,

$$f_{i,n}(t) = \begin{cases} f_{j,n-1}(t) & \text{for } 1 \leq t \leq n-1 \\ i & \text{for } t=n \end{cases} , \quad (7b)$$

where  $j$  is the argument minimizing (7a). Note that the range of minimization over  $j$ , constrained to the interval  $1 \leq j \leq i$ , guarantees the satisfaction of the monotonicity constraint (4).

The procedure is initialized for  $n=1$  by setting

$$D_{1,1} = d(g_R(1), g(1)) , \quad (8a)$$

and

$$D_{i,1} = \infty , \quad 2 \leq i \leq T_R , \quad (8b)$$

and is terminated when  $n=T$ . This initialization assures that the optimal curve ending at any point in the grid (including point  $B$ ) does start at point  $A$ , according to the global constraints of (3). Therefore, the optimal curve connecting point  $A$  to point  $B$  is found after  $T$  iterations, each requiring on the order of  $T_R$  operations described by (7) so that the total computational cost is  $O(TT_R)$ .

## 2.2 Matching Images

**2.2.1 DPW Problem Formulation** In extending the DTW algorithm to the alignment of images, our goal is to match the 2-dimensional reference image,  $G_R = \{g_R(x,y) : x \in Z^+, y \in Z^+, (x,y) \in L_{X_R,Y_R}, g_R(\cdot,\cdot) \in G \subset R^n\}$  to an elastically distorted test image,  $G = \{g(x,y) : x \in Z^+, y \in Z^+, (x,y) \in L_{X,Y}, g(\cdot,\cdot) \in G \subset R^n\}$ . Here an  $(x,y)$  pair describes pixel location by horizontal and vertical coordinates, and  $L_{N,M}$  denotes a rectangular discrete lattice, i.e., a set of pixels  $L_{N,M} = \{(x,y) \mid 1 \leq x \leq N, 1 \leq y \leq M\}$ . Figure 2 shows a simple example of  $G_R$  and  $G$ . This example is used to illustrate the definitions and the procedures described below.

The idea of planar warping is to map the test lattice to the reference one through a mapping function  $F$ :

$$\begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} = F \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} F_x(x,y) \\ F_y(x,y) \end{bmatrix}, \quad (9)$$

such that the distortion

-14-

$$D(G_R, G) = D = \sum_{x=1}^X \sum_{y=1}^Y d(g_R(\tilde{x}, \tilde{y}), g(x, y)) \quad (10)$$

$$\begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} = F \begin{bmatrix} x \\ y \end{bmatrix}$$

is minimal, subject to possible constraints like global boundary conditions:

$$F_x(1, y) = 1 ; \quad (11a)$$

$$F_x(X, y) = X_R ; \quad (11b)$$

$$F_y(x, 1) = 1 ; \quad (11c)$$

$$F_y(x, Y) = Y_R , \quad (11d)$$

and local monotonicity constraints, such as

$$\Delta F_{xx} = F_x(x+1, y) - F_x(x, y) \geq 0 ; \quad (12a)$$

$$\Delta F_{yy} = F_y(x, y+1) - F_y(x, y) \geq 0 . \quad (12b)$$

We denote by  $F$  the set of all admissible mappings that satisfy the above conditions. Although we limit the discussion in this appendix to constraints (11) and (12), the treatment of other kinds of constraints is similar.

**2.2.2 The General Approach** The complexity of the problem of finding the optimal warping function is exponential, namely  $O((X_R Y_R)^{XY})$ . This complexity can be reduced, as in the one-dimensional case, by generalizing the optimality principle. We will use the following definitions:



1. Define  $\Theta$  to be a set of  $N_T$  test sub-shapes  $\{\theta_n\}$ , where each test sub-shape is a set of pixels  $\{(x,y)\}$  satisfying the following conditions:

$$N_T \text{ is polynomial in } \{X,Y\}, \quad (13)$$

$$\theta_n \supset \theta_{n-1}, \quad 1 \leq n \leq N_T,$$

$$\Delta\theta_n \equiv \left\{ (x,y) \mid (x,y) \in \theta_n \text{ and } (x,y) \notin \theta_{n-1} \right\} \text{ has a natural mono-dimensional parametrization,}$$

$$\theta_{N_T} = L_{X,Y},$$

where  $\theta_0$  is the empty set. In particular, we choose  $\theta$  to be a set of  $Y$  rectangles,  $\theta_n = L_{X,n}$  (see Fig. 2),  $n=1, \dots, Y$ . In this case  $\Delta\theta_n$  are pixels of the  $n$ -th row.

2. Define  $\Phi$  to be a set of admissible warping sequences  $\Phi = \left\{ \phi_i \mid 1 \leq i \leq N_\Phi \right\}$ , where  $\phi_i$  is

a sequence of  $X$  reference pixels  $\phi_i = \left\{ (\tilde{x}_1^i, \tilde{y}_1^i), \dots, (\tilde{x}_x^i, \tilde{y}_x^i), \dots, (\tilde{x}_X^i, \tilde{y}_X^i) \right\}$  that meets the following conditions:

$$\phi_i \subset L_{X_R, Y_R}; \quad (14)$$

$$\tilde{x}_1^i = 1, \quad \tilde{x}_X^i = X_R;$$

$$\tilde{x}_{x+1}^i \geq \tilde{x}_x^i, \quad 1 \leq x \leq X.$$

This definition of the set  $\Phi$  depends on the particular choice of the set  $\Theta$  and the constraints (11a), (11b) and (12a).  $\Phi$  is constructed to contain all possible warping

-16-

sequences of each  $\Delta\theta_n$  that satisfy the constraints.

From this definition it is clear that for each  $i$ ,  $1 \leq i \leq N_\Phi$ , and for each  $n$ ,  $2 \leq n \leq Y-1$ ,

there exists  $F \in \mathbf{F}$  such that  $\begin{bmatrix} \tilde{x}_x^i \\ \tilde{y}_x^i \end{bmatrix} = F \begin{bmatrix} x \\ n \end{bmatrix}$  for  $1 \leq x \leq X$ . Also for each  $F \in \mathbf{F}$  and any  $n$ ,

$1 \leq n \leq Y$ , there exists  $\phi_i \in \Phi$  such that  $\begin{bmatrix} \tilde{x}_x^i \\ \tilde{y}_x^i \end{bmatrix} = F \begin{bmatrix} x \\ n \end{bmatrix}$  for  $1 \leq x \leq X$ . The cardinality of  $\Phi$  is

$$N_\Phi = O((X_R Y_R)^X).$$

3. Each sequence  $\phi_i \in \Phi$  determines a subset  $\Lambda_i \subset \Phi$  of sequences

$$\Lambda_i = \left\{ \phi_k : \tilde{y}_x^k \leq \tilde{y}_x^i, 1 \leq x \leq X \right\}. \quad (15)$$

Whenever we consider  $\phi_i$  to be a candidate warping sequence for the  $n$ -th row of the test image, the preceding  $(n-1)$ -th row can be matched only with a warping sequence in  $\Lambda_i$  in order to meet the vertical monotonicity condition (12b).

Figure 3 shows the concepts defined above, applied to the example of figure 2. In figure 3a the set  $\Theta$  is shown. The set  $\Phi$  includes in this case 16 sequences, shown in figure 3b. The corresponding  $\Lambda_i$  for each  $\phi_i \in \Phi$  is also given.

4. Denote by  $F_{i,n}$  a set of sub-mapping functions from the  $n$ -th test rectangle  $\theta_n$ ,  $1 \leq n \leq N_T$ , that satisfy the monotonicity conditions (12a) and (12b), boundary conditions (11a), (11b) and (11c), and match the  $n$ -th row of the test  $\Delta\theta_n$  with  $\phi_i$ :

$$\text{for any } F \in F_{i,n} \quad \begin{bmatrix} \tilde{x}_x^i \\ \tilde{y}_x^i \end{bmatrix} = F \begin{bmatrix} x \\ n \end{bmatrix} \quad 1 \leq x \leq X. \quad (16)$$

-17-

Using these definitions we are ready to describe the DPW algorithm.

In the  $n$ -th iteration of the algorithm,  $2 \leq n \leq Y$ , we assume that the optimal warpings of the  $(n-1)$ -th rectangle of the test image  $g(x,y)$ ,  $(x,y) \in \theta_{n-1}$ , that match the  $(n-1)$ -th test image row  $g(x,y)$ ,  $(x,y) \in \Delta\theta_{n-1}$  with the warping sequence  $\phi_i$  are known for  $1 \leq i \leq N_\phi$ . Each optimal warping is defined by a mapping  $F_{i,n-1} \in \mathbf{F}_{i,n-1}$  and a distortion  $D_{i,n-1}$ , such that

$$D_{i,n-1} = \min_{F \in \mathbf{F}_{i,n-1}} \sum_{x=1}^X \sum_{y=1}^{n-1} d(g_R(\tilde{x}, \tilde{y}), g(x,y)) ; \quad (17)$$

$$\begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} = F \begin{bmatrix} x \\ y \end{bmatrix}$$

$$F_{i,n-1} = \arg \min_{F \in \mathbf{F}_{i,n-1}} \sum_{x=1}^X \sum_{y=1}^{n-1} d(g_R(\tilde{x}, \tilde{y}), g(x,y)) . \quad (18)$$

$$\begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} = F \begin{bmatrix} x \\ y \end{bmatrix}$$

Now we can find the optimal warping of the  $n$ -th test rectangle,  $g(x,y)$ ,  $(x,y) \in \theta_n$ , that matches the  $n$ -th test image row to to the  $j$ -th warping sequence,  $g_R(x,y)$ ,  $(x,y) \in \phi_j$ :

$$D_{j,n} = \min_{F \in \mathbf{F}_{j,n}} \sum_{x=1}^X \sum_{y=1}^n d(g_R(\tilde{x}, \tilde{y}), g(x,y)) = \quad (19)$$

$$\begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} = F \begin{bmatrix} x \\ y \end{bmatrix}$$

$$= \min_{i \in \Lambda_j} \min_{F \in \mathbf{F}_{i,n-1}} \sum_{x=1}^X \sum_{y=1}^{n-1} d(g_R(\tilde{x}, \tilde{y}), g(x,y)) + \sum_{x=1}^X d(g_R(\tilde{x}_x, \tilde{y}_x^j), g(x,n)) =$$

$$\begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} = F \begin{bmatrix} x \\ y \end{bmatrix}$$

-18-

$$= \min_i D_{i,n-1} + \sum_{x=1}^X d(g_R(\tilde{x}_x^j, \tilde{y}_x^j), g(x, n)).$$

$\phi_i \in \Lambda_j$

The optimal mapping  $F_{j,n}$  is

$$F_{j,n}(x,y) = \begin{cases} F_{i,n-1}(x,y) & \text{for } (x,y) \in \theta_{n-1} \\ (\tilde{x}_x^j, \tilde{y}_x^j) & \text{for } (x,y) \in \Delta\theta_n \end{cases} \quad (20)$$

where  $i$  is the argument minimizing (19). Constraining the minimization in (19) only to those  $i$  such that  $\phi_i \in \Lambda_j$ , guarantees that the vertical monotonicity condition (12b) is satisfied. The horizontal monotonicity condition (12a), and the two boundary conditions (11a) and (11b) are satisfied through the definition of  $\phi_j$ .

To complete the  $n$ -th iteration, the optimal warping of the  $n$ -th test rectangle has to be found for every warping sequence  $\phi_j \in \Phi$ , thus requiring  $N_\Phi X$  operations.

The algorithm is initialized for  $n=1$  by setting

$$D_{1,i} = \sum_{x=1}^X d(g(x, 1)g_R(\tilde{x}_x^i, \tilde{y}_x^i)) \delta^{-1}(\tilde{y}_x^i - 1), \quad (21)$$

where  $\delta(\cdot)$  is the Kronecker delta function. This initialization guarantees the satisfaction of condition (11c). The algorithm is stopped after  $n=Y$ , when the optimal warpings  $F_{i,Y}$  are found for all  $i$  for which  $\Lambda_i = \Phi$ , thereby requiring a total of  $O(YXN_\Phi)$  computations. The global optimal warping function  $F_{\text{optimal}}$  minimizing (10) and satisfying (11) and (12) is chosen among these warpings as the one that produces the minimal distortion:

-19-

$$F_{optimal} = F_{j,Y}, \quad (22)$$

where

$$j = \arg \min_{i: \Lambda_i = \Phi} D_{i,Y}.$$

Constraining the minimization in (22) only to those  $i$  for which  $\Lambda_i = \Phi$ , guarantees satisfaction of the boundary condition (11d).

Figure 4 shows the values of  $D_{i,n}$  and  $F_{optimal}$  for the example of figure 2, using a quadratic distance measure  $d(g_R(\tilde{x}_x^j, \tilde{y}_x^j), g(x, n)) = (g_R(\tilde{x}_x^j, \tilde{y}_x^j) - g(x, n))^2$ .

**2.2.3 Constraining the Warping Problem** Even though applying the optimality principle reduces the complexity of the planar warping, the computation is still exponential. Therefore the algorithm is impractical for real-size images (since  $N_\Phi = O((Y_R X_R)^X)$ ). Further reduction of the computational complexity can be achieved in two different ways:

1. Finding a sub-optimal solution to the warping problem. Examples of sub-optimal procedures can be found in [3,4], where the images are divided into small sub-images, usually containing up to three rows of pixels. These sub-images were small enough, that finding a (local) optimal warping function is possible. The global solution, however, is not optimal, since the dependence across sub-images is neglected.
2. Redefining and simplifying the original warping problem.

The idea here is to limit the number of admissible warping sequences in  $\Phi$ , or, equivalently, constrain the class of admissible mappings  $F$  in such a way that an optimal solution to the constrained problem can be found in polynomial time. The

-20-

additional constraints used are not arbitrary, but instead reflect the geometric properties of the specific set of images being compared. For example, we can constrain the possible mappings to be of the form

$$\begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} = F \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} F_x(x,y) \\ F_y(y) \end{bmatrix}, \quad (23)$$

where the vertical distortion is independent of the horizontal position. In this case  $N_\Phi = O(X_R^X Y)$ , and the admissible warping sequences  $\phi \in \Phi$  are naturally grouped into  $Y_R$  subsets. The  $m$ -th subset,  $\lambda_m$  contains all those sequences  $\phi_i$  for which  $\tilde{y}_x^i = m, 1 \leq x \leq X$  (e.g., in figure 2, the set  $\Phi$  contains only four sequences,  $\Phi = \{\phi_1, \phi_5, \phi_{12}, \phi_{16}\}$ , and  $\lambda_1 = \{\phi_1, \phi_5\}$ ,  $\lambda_2 = \{\phi_{12}, \phi_{16}\}$ ). For all  $\phi_i \in \lambda_m$ ,  $\Lambda_i = \bigcup_{k=1}^m \lambda_k$ , i.e., the satisfaction of the vertical monotonicity condition is independent of a particular horizontal warping. This allows further reduction of computational complexity as follows. We define

$$\hat{D}_{m,n} \equiv \min_{j: \phi_j \in \lambda_m} D_{j,n}, \quad (24a)$$

and

$$\hat{F}_{m,n} = F_{i,n}, \quad (24b)$$

where  $i$  is the argument that minimizes (24a). The recursion relation (19) can now be rewritten in terms of these quantities as:

-21-

$$\begin{aligned}
\hat{D}_{m,n} &\equiv \min_{j: \phi_j \in \lambda_m} \min_{i \in \Lambda_j} \min_{F \in F_{i,n-1}} \sum_{x=1}^X \sum_{y=1}^{n-1} d(g_R(\tilde{x}, \tilde{y}), g(x, y)) + \min_{j: \phi_j \in \lambda_m} \sum_{x=1}^X d(g_R(\tilde{x}_x^j, \tilde{y}_x^j), g(x, n)) = \\
&\quad \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} = F \begin{bmatrix} x \\ y \end{bmatrix} \\
&= \min_{1 \leq k \leq m} \hat{D}_{k,n-1} + \Delta D_{m,n} \tag{25}
\end{aligned}$$

The second term of (25),  $\Delta D_{m,n} = \min_{j: \phi_j \in \lambda_m} \sum_{x=1}^X d(g_R(\tilde{x}_x^j, \tilde{y}_x^j), g(x, n))$ , is the distortion resulting from optimally aligning the  $n$ -th row of the test image to the  $m$ -th row of the reference image while satisfying both the horizontal monotonicity and the boundary conditions (11a), (11b) and (12a). This is equivalently a single-dimensional warping, as described in the previous section, and requires  $O(XX_R)$  calculations. Denote by  $f_{m,n}$  the optimal mapping that aligns the  $n$ -th row of the test image to the  $m$ -th row of the reference image constrained by (11a), (11b) and (12a) and minimizing  $\Delta D_{m,n}$ . Then

$$\hat{F}_{m,n} = \begin{cases} \hat{F}_{i,n-1} & \text{for } (x, y) \in \theta_{n-1} \\ f_{m,n} & \text{for } (x, y) \in \Delta\theta_n \end{cases}, \tag{26}$$

where  $i = \arg \min_{1 \leq k \leq m} \hat{D}_{k,n-1}$ .

The optimal mapping,  $F_{optimal} \in F$ , minimizing (10) and satisfying (11,12), is  $F_{optimal} = \hat{F}_{Y_R, Y}$ , and the complexity of its computation is only  $O(Y_R Y X_R X)$ ! Figure 5 shows  $\hat{D}_{m,n}$  and  $F_{optimal}$  obtained by applying the restricted approach to the problem of figure 2. Note that the solution obtained here is the same as the one obtained by the general approach shown in figure 4.

An important remaining question is what are the limitations of this restricted approach, as compared to the original one? Assumption (23) implies that a row of the test

image can be mapped to pixels that belong to a single row of the reference, i.e., a horizontal line in the test image will be mapped to some horizontal line in the reference image. This fact does not severely restrict the generality of the approach, since many kinds of distortion can be accounted for in this manner. For example, a straight line with a small but non-zero slope can be transformed into any straight or not straight line, excluding the line with zero slope. An example of a test image, for which the solution obtained by the restricted approach differs from that obtained by the general approach, is shown in figure 6.

The restricted formulation of the problem should reflect the geometry of the application. The restriction (23) discussed here is only one among many possibilities. For other restricted formulations it might be useful to design the sets  $\Theta$  and  $\Phi$  in a different manner. For example,  $\Theta$  can be the set of nested vertical rectangles  $\{R_{n,Y}: 1 \leq n \leq X\}$ ;  $\Phi$ , in this case, includes the warping sequences for test image columns similarly to (14), and the set of admissible mappings is restricted to contain functions  $F$  for which  $F_x(x,y)=F_x(x)$ . Generic description of the type of needed constraints is presented in appendix A.

### 3. Statistical Modeling Approach to Pattern Recognition

Another way of approaching the pattern recognition problem is by means of statistical modeling of the pattern source. The  $k$ -th class of patterns  $C_k$ ,  $k=1, \dots, N_c$ , is represented by a model, which is assumed to generate the  $k$ -th class patterns according to the probability distribution  $P(G | C_k)$ . Under this paradigm, the criterion that yields the minimal classification error is maximum a posteriori probability decoding: an



unclassified pattern  $G$  is assigned to the class  $C_k$  according to

$$C_k = \arg \max_{1 \leq n \leq N_c} P(C_n | G). \quad (27)$$

The term  $P(C_n | G)$  can be rewritten as

$$P(C_n | G) = \frac{P(G | C_n)P(C_n)}{P(G)}, \quad (28)$$

where  $P(G)$  is independent of  $C_n$  and therefore can be ignored. The prior class probability  $P(C_n)$  is generally attributed to higher level knowledge (e.g., syntactic knowledge). If such knowledge is not readily available, we usually assume a uniform class probability  $P(C_n) = \frac{1}{N_c}$ . Then the classification problem is that of maximizing the likelihood

$$P(G | C_n) \equiv P_n(G). \quad (29)$$

The computation of this likelihood is performed using the underlying stochastic model that represents the  $n$ -th class.

In the next subsection we describe a stochastic model called the "Hidden Markov model" frequently used to model temporal signals. We show that the statistical classification approach, using this model, generalizes the template matching paradigm based on DTW. Then we proceed to define a new stochastic model, that can both extend the HMM approach for planar signals, and generalize the template matching approach using DPW.

### 3.1 Hidden Markov Model

The HMM is a statistical model that is used to compute  $P_n(G)$  for temporal signals  $G = \{g(t) : 1 \leq t \leq T, g \in G \subset \mathbb{R}^n\}$  such as speech [4] [5] [6]. For simplicity we omit the class index  $n$ . The HMM is a composite statistical source, comprising a set of  $T_R$  sources, called states  $s = \{1, \dots, T_R\}$ . The  $i$ -th state,  $i \in s$ , is characterized by its probability distribution over  $G$ ,  $P_i(g)$ . At each time  $t$  only one of the states is active, emitting the observable  $g(t)$ . We denote the random variable, corresponding to the active state at time  $t$  by  $s(t)$ ,  $s(t) \in s$ . The joint probability distribution (for real-valued  $g$ ) or discrete probability mass (for  $g$  being a discrete variable)  $P(s(t), g(t))$  for  $t > 1$  is characterized by the following property:

$$\begin{aligned} P(s(t), g(t) \mid s(1:t-1), g(1:t-1)) &= P(s(t) \mid s(t-1)) P(g(t) \mid s(t)) \equiv \\ &= P(s(t) \mid s(t-1)) P_{s(t)}(g(t)), \end{aligned} \quad (30)$$

where  $s(1:t-1)$  stands for the sequence  $\{s(1), \dots, s(t-1)\}$ , and  $g(1:t-1) = \{g(1), \dots, g(t-1)\}$ .

We denote by  $a_{ij}$  the transition probability  $P(s(t)=j \mid s(t-1)=i)$ , and by  $\pi_i$ , the probability of state  $i$  being active at  $t=1$ ,  $\pi_i = P(s(1)=i)$ .

The probability of the entire sequence of states  $S \equiv s(1:T)$  and observations  $G = g(1:T)$  can be expressed as

$$P(G, S) = \pi_{s(1)} P_{s(1)}(g(1)) \prod_{t=2}^T a_{s(t-1)s(t)} P_{s(t)}(g(t)). \quad (31)$$

The interpretation of equations (30) and (31) is that the observable sequence  $G$  is generated in two stages: first, a sequence  $S$  of  $T$  states is chosen according to the Markovian distribution parametrized by  $\{a_{ij}\}$  and  $\{\pi_i\}$ ; then each one of the states  $s(t)$ ,  $1 \leq t \leq T$ , in  $S$  generates an observable  $g(t)$  according to its own memoryless distribution  $P_{s(t)}$ , forming the observable sequence  $G$ . This model is called a *hidden* Markov model, because the state sequence  $S$  is not given in most applications, and only the observation sequence  $G$  is known. We can estimate the most probable state sequence  $\hat{S}$ , given the observation  $G$ , as

$$\hat{S} = \underset{S}{\operatorname{argmax}} P(S | G) = \underset{S}{\operatorname{argmax}} P(G, S). \quad (32)$$

Then the likelihood (29) of the sequence of observations is approximated by

$$\hat{P}(G) = P(G, \hat{S}), \quad (33)$$

i.e., instead of the sum

$$P(G) = \sum_S P(G, S), \quad (34)$$

only the maximal term is taken into account. This approximation is computationally economical, and has been shown, both experimentally and theoretically<sup>[7]</sup>, to be valid, i.e., to have a vanishingly small approximation error.

The problem of finding  $\hat{S}$  and  $\hat{P}(G)$  can be restated as that of minimizing

$$L \equiv \sum_{t=1}^T -\log P_{s(t)}(g(t)) + \sum_{t=2}^T -\log a_{s(t-1)s(t)} - \log \pi_{s(1)} = D + C, \quad (35)$$

-26-

over all possible state sequences  $S$ . The problem of minimizing  $L$  is of exponential complexity, since there exist  $T^{T_R}$  possible state sequences, but it can be solved in polynomial time using a dynamical programming approach (similarly to description of Section 2.1). It is useful to understand this similarity: a state sequence  $S$  defines a mapping from the observation time scale  $1 \leq t \leq T$  to the active state at time  $t$ ,  $1 \leq s(t) \leq T_R$ , that corresponds to the reference time scale  $\tilde{t}$  in the DTW approach. The first term in (35),  $D = \sum_{t=1}^T -\log P_{s(t)}(g(t))$ , provides a distortion measure, as in (2). For example, for a

Gaussian HMM, where  $P_i(g) = \frac{1}{2\pi} \exp[ -\|g - \mu_i\|^2 ]$ ,  $D = \sum_{t=1}^T \|g - \mu_{\tilde{t}}\|^2 + \text{const}$ , where

$\tilde{t} = s(t)$ . The penalty term in (35),  $C = -\sum_{t=1}^T \log a_{s(t-1)s(t)} - \log \pi_{s(1)}$ , generalizes the global

and the local constraints of equations (3) and (4) of DTW. A particular case of this model, called a *left-to-right* HMM, is especially useful for speech modeling and recognition. In this case  $a_{ij} = 0$  for  $j < i$ , and  $\pi_1 = 1$ . This type of model provides an infinite penalty to state sequences that do not start with  $s_1 = 1$ , and for which the monotonicity condition  $s(t+1) \geq s(t)$  does not hold. If in addition the absorbing state  $s(T)$  is constrained to be the last state of the model  $s(T) = T_R$ , the minimization (35) is, in effect, performed only among those state sequences that correspond to mappings satisfying conditions that are equivalent to (3) and (4). The only difference between the minimization problem defined by (2), (3) and (4) and this one is the non-zero penalty term in (35). The optimality principle can be applied to the minimization (35) in a manner similar to DTW as described in section 2.1.2.

This statistical description not only provides a formal interpretation of the heuristic warping procedure and aids its understanding, but also enables natural integration with

higher-level syntactical knowledge.

### 3.2 The Two-Dimensional Case: Planar HMM

In this section we describe a statistical model for  $P_n(G)$ , when  $G$  is a planar image,  $G = \{g(x,y):(x,y) \in L_{X,Y}, g \in G\}$ . We call this model the "Planar HMM" (PHMM) and design it not only to extend the conventional HMM to the two-dimensional case, but also to provide a statistical interpretation and generalization of the DPW approach.

The PHMM is a composite source, comprising a set,  $s$ , of  $N=X_R Y_R$  states  $s = \{(\tilde{x}, \tilde{y}), 1 \leq \tilde{x} \leq X_R, 1 \leq \tilde{y} \leq Y_R\}$ . Each state in  $s$  is a stochastic source characterized by its probability density  $P_{\tilde{x}, \tilde{y}}(g)$  over the space of observations  $g \in G$ . It is convenient to think of the states of the model as being located on a rectangular lattice  $L_{X_R Y_R}$ , corresponding to the reference lattice of DPW. Similarly to the conventional HMM, only one state is active in the generation of the  $(x,y)$ -th image pixel  $g(x,y)$ . We denote by  $s(x,y) \in s$  the active state of the model that generates  $g(x,y)$ . The joint distribution governing the choice of active states and image values has the following Markovian property (see figure 7):

$$\begin{aligned} P(g(x,y), s(x,y) \mid g(1:X, 1:y-1), g(1:x-1,y), s(1:X, 1:y-1), s(1:x-1,y)) &= \quad (36) \\ &= P(g(x,y) \mid s(x,y)) P(s(x,y) \mid s(x-1,y), s(x,y-1)) = \\ &= P_{s(x,y)}(g(x,y)) P(s(x,y) \mid s(x-1,y), s(x,y-1)) = \end{aligned}$$

where  $g(1:X, y-1) \equiv \{g(x,y):(x,y) \in R_{X,y-1}\}$ ,  $g(1:x-1,y) \equiv \{g(1,y), \dots, g(x-1,y)\}$ , and  $s(1:X, 1:y-1)$ ,  $s(1:x-1,y)$  are the active states involved in generating  $g(1:X, y-1)$ ,  $g(1:x-1,y)$ , respectively (see figure 6). Using property (36), the joint likelihood of the

-28-

image  $G=g(1:X, 1:Y)$  and the state image  $S=s(1:X, 1:Y)$  can be written as

$$P(G, S) = \prod_{x=1}^X \prod_{y=1}^Y P_{s(x,y)}(g(x,y)) \quad (37)$$

$$\pi_{s(1,1)} \prod_{x=2}^X a_{s(x-1,1), s(x,1)}^H \prod_{y=2}^Y a_{s(1,y-1), s(1,y)}^V \prod_{y=2}^Y \prod_{x=2}^X A_{s(x-1,y), s(x,y-1), s(x,y)}$$

where

$$A_{(i,j),(k,l),(m,n)} \equiv P(s(x,y)=(m,n) \mid s(x-1,y)=(i,j), s(x,y-1)=(k,l)), \quad (38)$$

$$a_{(i,j),(m,n)}^H \equiv P(s(x,1)=(m,n) \mid s(x-1,1)=(i,j)),$$

$$a_{(k,l),(m,n)}^V \equiv P(s(1,y)=(m,n) \mid s(1,y)=(k,l)),$$

$$\pi_{ij} \equiv P(s(1,1)=(i,j)).$$

Similarly to HMM, (37) suggests that an image  $G$  is generated by the PHMM in two successive stages: in the first stage the state matrix  $S$  is generated according to the Markovian probability distribution parametrized by  $\{A\}$ ,  $\{a^H\}$ ,  $\{a^V\}$ , and  $\{\pi\}$ . In the second stage, the image value in the  $(x,y)$ -th pixel is produced independently from other pixels according to the distribution of the  $s(x,y)$ -th state  $P_{s(x,y)}(g)$ . As in HMM, the state matrix  $S$  in most of the applications is not known, only  $G$  is given. The state matrix  $\hat{S}$  that best explains the observable  $G$  can be estimated as in (32) by  $\hat{S} = \underset{S}{\operatorname{argmax}} P(G, S)$ , and then observation likelihood  $P(G)$  is approximated as  $\hat{P}(G) = P(G, \hat{S})$ .

-29-

Therefore, the problem of finding  $\hat{S}$  and  $\hat{P}(G)$  is that of minimizing

$$\begin{aligned}
 L \equiv & \sum_{x=1}^X \sum_{y=1}^Y -\log P_{s(x,y)}(g(x,y)) + \\
 & -\log \pi_{s(1,1)} - \sum_{x=2}^X \log a_{s(x-1,1),s(x,1)}^H - \sum_{y=2}^Y -\log a_{s(1,y-1),s(1,y)}^V - \sum_{x=2}^X \sum_{y=2}^Y A_{s(x-1,y),s(x,y-1),s(x,y)} = \\
 & = D + C
 \end{aligned} \tag{39}$$

over all possible state matrices  $S$ . Again, the problem is of exponential complexity, since there are  $(X_R Y_R)^{XY}$  different state matrices. This complexity can be reduced, as with DPW, by applying the optimality principle and by restricting the model. The similarity between the problem of finding the most probable state matrix in PHMM and DPW can be shown as follows: the states of the PHMM correspond to the pixels of the reference image, and therefore the active state matrix  $S$  corresponds to the mapping  $F$  of DPW. The first term in  $L$ ,  $D = \sum_{x=1}^X \sum_{y=1}^Y -\log P_{s(x,y)}(g(x,y))$  is equivalent to the distortion measure  $D$  of DPW. The second term,  $C$ , generalizes constraints (11), (12), and (23). In particular, by restricting the PHMM parameter values to be

$$\pi_{(1,1)} = 1 ; \tag{40}$$

$$a_{(i,j),(m,n)}^H \neq 0 \text{ only for } m \geq i \text{ and } j = n = 1 ;$$

$$a_{(k,l),(m,n)}^V \neq 0 \text{ only for } n \geq l \text{ and } k = m = 1 ;$$

$$A_{(i,j),(k,l),(m,n)} \neq 0 \text{ only for } i \leq m \text{ and for } l \leq n ,$$

the active state matrix  $S$  that minimizes (38) must satisfy conditions equivalent to (11),

-30-

(12a) and (12c). The PHMM constrained by (40) can be referred to as the *left-to-right bottom-up* PHMM, since it doesn't allow for "foldovers" in the state images.

The other boundary conditions (12b) and (12d) can be imposed on  $\hat{S}$  by restricting the values of  $s(x, Y)$ ,  $1 \leq x \leq X$  and  $s(X, y)$ ,  $1 \leq y \leq Y$ ,

$$\begin{aligned} s(x, Y) &\in \left\{ (\tilde{x}, Y_R) \mid 1 \leq \tilde{x} \leq X_R \right\}; \\ s(X, y) &\in \left\{ (X_R, \tilde{y}) \mid 1 \leq \tilde{y} \leq Y_R \right\}. \end{aligned} \quad (41)$$

**3.2.1 Constraining the parameters of PHMM** In this section we describe the ways of constraining the values of transition probabilities  $\{A_{(i,j),(k,l),(m,n)}\}$  in order to reduce the complexity of the problem of finding  $\hat{S}$  and  $\hat{P}(G)$  to polynomial, similarly to the additional constraints on DPW discussed in appendix A, and section 2.2.3.

For the problem of finding  $\hat{S}$  and  $\hat{P}(G)$  to be solved in polynomial time, there should exist a grouping of the set  $s$  of states of the model into  $N_G$  subsets of states  $\gamma_p$ ,

$s = \bigcup_{p=1}^{N_G} \gamma_p$ . These subsets do not have to be mutually exclusive, and can share states.

Two examples of such groupings are shown in figure 8. The number of subsets,  $N_G$ , should be polynomial in the dimensions of the model  $X_R, Y_R$ . The probabilities  $\{A_{(i,j),(k,l),(m,n)}\}$  should satisfy the two following constraints with respect to such grouping:

$$A_{(i,j),(k,l),(m,n)} \neq 0 \text{ only if there exists } p, 1 \leq p \leq N_G, \text{ such that } (i,j), (m,n) \in \gamma_p. \quad (42)$$

Condition (42) means that the left neighbor of the state  $(m,n)$  in the state matrix  $S$



-31-

must be a member of the same group  $\gamma_p$  as  $(m,n)$ . The second constraint is:

$$A_{(i,j),(k,l),(m,n)} = A_{(i,j),(k,l_1),(m,n)} \text{ if there exists } p, 1 \leq p \leq N_G, \text{ such that} \quad (43a)$$

$$(k,l), (k,l_1) \in \gamma_p, .$$

$$\frac{A_{(i,j),(k,l),(m,n)}}{A_{(i,j),(k,l_1),(m,n)}} = \frac{A_{(i,j),(k,l),(m,n_1)}}{A_{(i,j),(k,l_1),(m,n_1)}} = \frac{A_{(i,j_1),(k,l),(m,n)}}{A_{(i,j_1),(k,l_1),(m,n)}} = K(r,r_1,p) \quad (43b)$$

if there exist  $p, 1 \leq p \leq N_G$ , such that  $(i,j), (i,j_1), (m,n), (m,n_1) \in \gamma_p$ , and where  $(k,l) \in \gamma_r, (k,l_1) \in \gamma_{r_1}$ .

The condition (43) makes the penalty term  $C$  independent of the horizontal warping.

In the case when (42) and (43) hold for a particular grouping, the nonzero transition probabilities  $A_{(i,j),(k,l),(m,n)}$  can be factorized into

$$A_{(i,j),(k,l),(m,n)} = A_{(i,j),(m,n)}^{TP} \times \alpha_{rp}, \quad (44)$$

where

$$A_{(i,j),(m,n)}^{TP} = P(s(x,y)=(m,n) \mid s(x-1,y)=(i,j), s(x,y), s(x-1,y) \in \gamma_p, s(x,y-1) \in \gamma_r), \quad (45)$$

and

$$\alpha_{rp} = P(s(x,y), s(x-1,y) \in \gamma_p \mid s(x,y-1) \in \gamma_r). \quad (46)$$

The ratio  $K(p,r,r_1)$  of Eq. (43b) can be expressed as  $\frac{\alpha_{rp}}{\alpha_{r,p}}$ . Using this equivalent representation of transition probabilities (given by equations (44-46)), a convenient description of PHMM can be derived. Each subset  $\gamma_p$  of PHMM can be considered as a one-dimensional HMM, comprising the states  $\{(\tilde{x}, \tilde{y}) \mid (\tilde{x}, \tilde{y}) \in \gamma_p\}$ , with transition

-32-

probabilities among those states  $A_{(i,j),(m,n)}^p$  of equation (45), and the respective observation probabilities. The whole PHMM can now be represented as a collection of such subsets, with a Markovian probability of transitions between the subsets defined by  $\alpha_{pr}$  of equation (46). This equivalent representation, illustrated in figure 9, suggests an iterative algorithm for computing the state matrix  $\hat{S}$  and  $\hat{P}(G)$  in polynomial time, similarly to DPW case of section 2.2.3. Denote by  $L_{p,n}$  the local cost, related to the probability that the  $n$ -th raw of the image  $G$  was generated by a single-dimensional HMM corresponding to the subset  $\gamma_p$ , and by  $\hat{S}_{p,n}$  the corresponding state sequence:

$$L_{p,n} = \min_{S_{p,n}} -\log P(g(1:X,n) \mid S(1:X,n) = S_{p,n}), \quad (47)$$

$$\hat{S}_{p,n} = \operatorname{argmin}_{S_{p,n}} -\log P(g(1:X,n) \mid S(1:X,n) = S_{p,n}). \quad (48)$$

This cost can be calculated in a polynomial time using the Viterbi algorithm, since this is a single-dimensional case. After all the local costs  $L_{p,n}$  have been calculated for  $1 \leq n \leq Y$ ,  $1 \leq p \leq N_G$ , the global cost  $L_{global} = -\log \hat{P}(G)$  and the optimal state matrix  $\hat{S}$  are found using the Viterbi algorithm for the single-dimensional HMM defined by a set of  $N_G$  states (the subsets  $\gamma_p$  of the PHMM), transition probabilities between these states ( $\alpha_{pr}$  of Eq. 46), and the observation probabilities given by  $\exp[-L_{p,n}]$ . The algorithm is illustrated in figure 10.

Although conditions (42),(43) are hard to check in practice since any possible grouping of the states has to be considered, they can be effectively used in constructive mode, i.e., choosing one particular grouping, and then imposing the

-33-

constraints (42) and (43) on the probabilities  $\{A_{(i,j),(k,l),(m,n)}\}$  with respect to this grouping. For example, if we choose  $\gamma_p = \{(\tilde{x}, \tilde{y}) \mid 1 \leq \tilde{x} \leq X_R, \tilde{y} = p\}$ ,  $1 \leq p \leq Y_r$ , then the constraints (42),(43) transform to

$$A_{(i,j),(k,l),(m,n)} \neq 0 \text{ only for } j = n, \quad (49)$$

and,

$$A_{(i,j),(k,l),(m,n)} = A_{(i,j),(k_1,l),(m,n)} \text{ for } 1 \leq k_1, k \leq X_R, \quad (50)$$

$$\frac{A_{(i,j),(k,l),(m,n)}}{A_{(i,j),(k_1,l),(m,n)}} = \frac{A_{(i,j),(k,l),(m,n)}}{A_{(i,j),(k_1,l),(m,n)}} = \frac{A_{(i,j),(k,l),(m_1,n)}}{A_{(i,j),(k_1,l),(m_1,n)}},$$

equivalently to the restriction imposed on DPW by (23).

The constraints (42) and (43) can be trivially changed by applying a coordinate transformation.

#### 4. Experimental Results

The PHMM approach was tested on a writer-independent isolated handwritten digit recognition application. The data we used in our experiments was collected from 12 subjects (6 for training and 6 for test). The subjects were each asked to write 10 samples of each digit. Each sample was written in a fixed-size box, therefore the samples were naturally size-normalized and centered. Figure 11 shows the 100 samples written by one of the subjects. Each sample in the database was represented by a 16x16 binary image.

-34-

Each character class (digit) was represented by a single PHMM, satisfying (49) and (50). Each PHMM had a *strictly* left-to-right bottom-up structure, where the state matrix  $S$  was restricted to contain every state of the model, i.e., states could not be skipped. All models had the same number of states. Each state was represented by its own binary probability distribution, i.e., the probability of a pixel being 1 (black) or 0 (white). We estimated these probabilities from the training data with the following generalization of the Viterbi training algorithm.<sup>[8]</sup> For the initialization we uniformly divided each training image into regions corresponding to the states of its model. The initial value of  $P_i(g=1)$  for the  $i$ -th state was obtained as a frequency count of the black pixels in the corresponding region over all the samples of the same digit. Each iteration of the algorithm consisted of two stages: first, the samples were aligned with the corresponding model, by finding the best state matrix  $\hat{S}$ . Then, a new frequency count for each state was used to update  $P_i(1)$ , according to the obtained alignment. We noticed that the training procedure converged usually after 2-4 iterations, and in all the experiments the algorithm was stopped at the 10th iteration. The recognition was performed as explained in section 3: The test sample was assigned to the class  $k$  for which  $\hat{P}_k(G)$  was maximal.

Table 1 shows the number of errors in the recognition of the training set and the test set for different sizes of the models.

Number of states $X_R = Y_R$	Recognition Errors	
	Training	Test
6	78	82
8	36	50
9	35	48
10	26	32
11	21	38
12	18	42
16	36	64

TABLE 1. Number of errors in the recognition of the training set and the test set for different size of the models (out of 600 trials in both cases)

It is worth noting the following two points. First, the test error shows a minimum for  $X_R = Y_R = 10$  of 5%. By increasing or decreasing the number of states this error increases. This phenomenon is due to the following:

1. The typical under/over parametrization behavior.
2. Increasing the number of states closer to the size of the modeled images reduces the flexibility of the alignment procedure, making this a trivial uniform alignment when  $X_R = Y_R = 16$ .

Also, the training error decreases monotonically with increasing number of states up to  $X_R = Y_R = 16$ . This is again typical behavior for such systems, since by increasing the

number of states, the number of model parameters grows, improving the fit to the training data. But when the number of states equals the dimensions of the sample images,  $X_R=Y_R=16$ , there is a sudden significant increase in the training error. This behavior is consistent with point (2) above.

Figure 12 shows three sets of models with different numbers of states. The states of the models in this figure are represented by squares, where the grey level of the square encodes the probability  $P(g=1)$ . The  $(6\times 6)$  state models have a very coarse representation of the digits, because the number of states is so small. The  $(10\times 10)$  state models appear much sharper than the  $(16\times 16)$  state models, due to their ability to align the training samples.

This preliminary experiment shows that eliminating elastic distortions by the alignment procedure discussed above plays an important role in the task of isolated character recognition, improving the recognition accuracy significantly. Note that the simplicity of this task does not stress the full power of the PHMM representation, since the data was isolated, size-normalized, and centered. We expect that the advantages of this approach will be even more prominent in harder tasks, such as cursive/connected hand-writing, recognition with grammatical constraints, noisy images, etc..

## 5. Summary and Discussion

In this appendix we demonstrated how the DTW algorithm and HM modeling, extensively used for speech recognition, can be generalized to OCR. We found two key problems in this generalization:

1. Applying the optimality principle in the planar case is not trivial, since the two

-37-

dimensions of an image cannot be treated separately. In order to use the optimality principle here, the set of all possible warping sequences satisfying horizontal constraints must be defined. For the  $n$ -th row of the test image every such sequence has to be considered as a candidate warping. The vertical constraints are taken into account by limiting the set of possible warping sequences of the previous  $(n-1)$ -th row. In this way the complexity of computation was reduced from  $O((X_R Y_R^{XY}))$  to  $O(YX(Y_R X_R)^X)$ .

2. Although applying the optimality principle reduces the computational complexity, it still remains exponential in the dimensions of the image. We show that by restricting the original warping problem by limiting the class of possible distortions (for example, assuming that the vertical distortion is independent of a horizontal position), we can reduce the computational complexity dramatically, and find the optimal solution to the restricted problem in linear time  $O(XY X_R Y_R)$ .

A statistical model (the planar hidden Markov model - PHMM) was developed to provide a probabilistic formulation to the planar warping problem. This model, on one hand, generalizes the single-dimensional HMM to the planar case, and on the other extends the DPW approach. The restricted formulation of the warping problem corresponds to PHMM with constrained transition probabilities. The PHMM approach was tested on an isolated, hand-written digit recognition application, yielding 95% digit recognition. Further analysis of the results indicate that even in a simple case of isolated characters, the elimination of planar distortions enhances recognition performance significantly. We expect that the advantage of this approach will be even more valuable in harder tasks, such as cursive writing recognition/spotting, for which an effective solution using the current available techniques has not yet been found.

## Figure Captions

Figure 1: Time-time grid. Abscissa: test time scale  $1 \leq t \leq T$ . Ordinate: reference time scale  $1 \leq t \leq T_R$ . Any monotonically increasing curve connecting point  $A$  to point  $B$  corresponds to a mapping  $f \in \mathcal{f}$ .

Figure 2: Example of warping problem.  $G_R$  is a  $2 \times 2$  reference image, and  $G$  is a  $3 \times 3$  test image. Inside each pixels are shown its  $(x,y)$  coordinates. The value of the image  $g(x,y)$  is encoded by texture, as shown.

Figure 3: Illustration of the definitions of  $\Theta$ ,  $\Phi$ , and  $\Lambda$  for the example of figure 2.

Figure 4: Illustration of the two-dimensional warping algorithm on the example of figure 2. The table shows the values of  $D_{i,n}$  for  $1 \leq i \leq 16$  and  $1 \leq n \leq 3$ , calculated according to the DPW algorithm. The optimal value of  $D$  is  $D=0$ , and the corresponding  $F_{optimal}$  is shown.

Figure 5: Illustration of the constrained DPW algorithm for the example of figure 2. The table shows the values of  $\hat{D}_{k,n}$  for  $1 \leq k \leq 2$  and  $1 \leq n \leq 3$ . In this case the obtained solution is the same as in figure 4.

Figure 6: Example of a test image  $G$  for which the optimal mapping obtained according to the general DPW formulation differs from the one obtained according to the restricted formulation.

Figure 7: Illustration of the planar Markov property. The probability of a state in the light grey pixel given the states of all the dark grey pixels in (a) equals the probability



of a state in the light grey pixel given the states of only two dark pixels in (b).

Figure 8: two groupings of the  $4 \times 4$  PHMM states into subsets.

- a. Here the set of states is divided into 4 mutually exclusive subsets, each contains states of one raw only.
- b. The same set of states are grouped into 7 subsets.

Figure 9: Equivalent representation of constrained PHMM, for the grouping of figure 8a.

Figure 10: Illustration of the algorithm for the case of figure 8a.

- a. First, the local costs are computed using Viterbi algorithm
- b. The global solution is found using Viterbi algorithm with the local costs.

Figure 11: The 100 samples of the digits from one subject.

Figure 12: The digit models obtained by training, for different number of states. The grey level in these images encodes the value of  $P(g=1)$  for each state.

## 6. Appendix A: Properties of the constraints

Changing the choice of sub-shape set  $\Theta$ , and changing the set of admissible mappings  $\Phi$  accordingly, is equivalent to coordinate transformation. The example discussed in the end of section 2.2.3, corresponds to such simple coordinate transformation, exchanging the roles of the vertical and the horizontal coordinates. In what follows we discuss a generic description of the constraints on the set  $\Phi$ , keeping the set  $\Theta$  fixed for  $\theta_n = L_{X,n}$ .

For the computational complexity of DPW process to be polynomial in the sizes of the images, there should exist a grouping of the set of admissible warping sequences (defined by  $F$ ) into  $N_G$  mutually exclusive subsets,  $\Phi = \bigcup_{k=1}^{N_G} \lambda_k$ . The number of subsets  $N_G$  should be polynomial in the sizes of the images  $\{X, Y, X_R, Y_R\}$ , and this grouping should fulfill the following conditions:

1. For  $1 \leq k \leq N_G$ , if  $\phi_i \in \lambda_k$  and  $\phi_j \in \lambda_k$  then  $\Lambda_i = \Lambda_j \equiv \Lambda^k$ .
2. For  $1 \leq k \leq N_G$ ,  $\Lambda^k$  can be expressed as a union of some subsets  $\lambda_j$ , i.e. for any  $k$ ,  $1 \leq k \leq N_G$ , there exist the indices  $\{k_1, k_2, \dots, k_{N_k}\}$ , such that  $\Lambda^k = \bigcup_{i=1}^{N_k} \lambda_{k_i}$ .

It is clear that the example (23) discussed in section 2.2.3, satisfy these conditions. The analysis of the general case described by conditions 1,2 above is similar to the analysis of the example (23) given in Eq. (24-26), and is therefore omitted.

## REFERENCES

1. R. Chellappa, S. Chatterjee, "Classification of textures Using Gaussian Markov Random Fields," *IEEE Transactions on ASSP* , Vol. 33, No. 4, pp. 959-963, August 1985.
2. H. Derin, H. Elliot, "Modeling and Segmentation of Noisy and Textured Images Using Gibbs Random Fields," *IEEE Transactions on PAMI* , Vol. 9, No. 1 pp. 39-55, January 1987.
3. R. Bellman, *Dynamic Programming*," Princeton, NJ: Princeton University Press, 1957
4. C.-H. Lee, L. R. Rabiner, R. Pieraccini, J. G. Wilpon, "Acoustic Modeling for Large Vocabulary Speech Recognition," *Computer Speech and Language*, 1990, No. 4, pp. 127-165.
5. J. G. Wilpon, L. R. Rabiner, C.-H. Lee, E. R. Goldman, "Automatic Recognition of Keywords in Unconstrained Speech Using Hidden Markov Models," *IEEE Trans. on ASSP*, Vol. 38, No. 11, pp 1870-1878, November 1990.
6. R. Pieraccini, E. Levin, "Stochastic Representation of Semantic Structure for Speech Understanding," *Proceedings of EUROSPEECH 91*, Vol.2, pp. 383-386, Genova, September 1991.

7. N. Merhav and Y. Ephraim, "Maximum likelihood hidden Markov modeling using a dominant sequence of states," accepted for publication in *IEEE Transaction on ASSP*.
8. F. Jelinek, "Continuous Speech Recognition by Statistical Methods," *Proceedings of IEEE*, vol. 64, pp. 532-556, April 1976.



-44-

$G =$

$(1,3)$	$(2,3)$	$(3,3)$
$(1,2)$	$(2,2)$	$(3,2)$
$(1,1)$	$(2,1)$	$(3,1)$

$G_{\pi} =$

$(1,2)$	$(2,2)$
$(1,1)$	$(2,1)$





	= 0.0
	= 0.2
	= 0.5
	= 1.0

Fig 2

-45-

$$\begin{aligned}\theta_1 &= \begin{array}{|c|c|c|} \hline (1, 1) & (2, 1) & (3, 1) \\ \hline \end{array} \leftarrow \Delta\theta_1 \\ \theta_2 &= \begin{array}{|c|c|c|} \hline (2, 1) & (2, 2) & (2, 3) \\ \hline (1, 1) & (2, 1) & (3, 1) \\ \hline \end{array} \begin{array}{l} \leftarrow \Delta\theta_2 \\ \leftarrow \Delta\theta_1 \end{array} \\ \theta_3 &= \begin{array}{|c|c|c|} \hline (3, 1) & (3, 2) & (3, 3) \\ \hline (2, 1) & (2, 2) & (2, 3) \\ \hline (1, 1) & (2, 1) & (3, 1) \\ \hline \end{array} \begin{array}{l} \leftarrow \Delta\theta_3 \\ \leftarrow \Delta\theta_2 \\ \leftarrow \Delta\theta_1 \end{array}\end{aligned}$$

FIGURE 3a

$\phi_1$	<table border="1"><tr><td>(1, 1)</td><td>(1, 1)</td><td>(2, 1)</td></tr></table>	(1, 1)	(1, 1)	(2, 1)	$\Lambda_1 = \{\phi_1, \phi_5\}$
(1, 1)	(1, 1)	(2, 1)			
$\phi_2$	<table border="1"><tr><td>(1, 1)</td><td>(1, 1)</td><td>(2, 2)</td></tr></table>	(1, 1)	(1, 1)	(2, 2)	$\Lambda_2 = \{\phi_1, \phi_2, \phi_5, \phi_6\}$
(1, 1)	(1, 1)	(2, 2)			
$\phi_3$	<table border="1"><tr><td>(1, 1)</td><td>(1, 2)</td><td>(2, 1)</td></tr></table>	(1, 1)	(1, 2)	(2, 1)	$\Lambda_3 = \{\phi_1, \phi_3, \phi_5, \phi_7\}$
(1, 1)	(1, 2)	(2, 1)			
$\phi_4$	<table border="1"><tr><td>(1, 1)</td><td>(1, 2)</td><td>(2, 2)</td></tr></table>	(1, 1)	(1, 2)	(2, 2)	$\Lambda_4 = \{\phi_1, \phi_2, \phi_3, \phi_4, \phi_5, \phi_6, \phi_7, \phi_8\}$
(1, 1)	(1, 2)	(2, 2)			
$\phi_5$	<table border="1"><tr><td>(1, 1)</td><td>(2, 1)</td><td>(2, 1)</td></tr></table>	(1, 1)	(2, 1)	(2, 1)	$\Lambda_5 = \Lambda_1$
(1, 1)	(2, 1)	(2, 1)			
$\phi_6$	<table border="1"><tr><td>(1, 1)</td><td>(2, 1)</td><td>(2, 2)</td></tr></table>	(1, 1)	(2, 1)	(2, 2)	$\Lambda_6 = \Lambda_2$
(1, 1)	(2, 1)	(2, 2)			
$\phi_7$	<table border="1"><tr><td>(1, 1)</td><td>(2, 2)</td><td>(2, 1)</td></tr></table>	(1, 1)	(2, 2)	(2, 1)	$\Lambda_7 = \Lambda_3$
(1, 1)	(2, 2)	(2, 1)			
$\phi_8$	<table border="1"><tr><td>(1, 1)</td><td>(2, 2)</td><td>(2, 2)</td></tr></table>	(1, 1)	(2, 2)	(2, 2)	$\Lambda_8 = \Lambda_4$
(1, 1)	(2, 2)	(2, 2)			
$\phi_9$	<table border="1"><tr><td>(1, 2)</td><td>(1, 1)</td><td>(2, 1)</td></tr></table>	(1, 2)	(1, 1)	(2, 1)	$\Lambda_9 = \Lambda_1 \cup \{\phi_9, \phi_{13}\}$
(1, 2)	(1, 1)	(2, 1)			
$\phi_{10}$	<table border="1"><tr><td>(1, 2)</td><td>(1, 1)</td><td>(2, 2)</td></tr></table>	(1, 2)	(1, 1)	(2, 2)	$\Lambda_{10} = \Lambda_2 \cup \{\phi_9, \phi_{10}, \phi_{13}, \phi_{14}\}$
(1, 2)	(1, 1)	(2, 2)			
$\phi_{11}$	<table border="1"><tr><td>(1, 2)</td><td>(1, 2)</td><td>(2, 1)</td></tr></table>	(1, 2)	(1, 2)	(2, 1)	$\Lambda_{11} = \Lambda_3 \cup \{\phi_9, \phi_{13}, \phi_{11}, \phi_{15}\}$
(1, 2)	(1, 2)	(2, 1)			
$\phi_{12}$	<table border="1"><tr><td>(1, 2)</td><td>(1, 2)</td><td>(2, 2)</td></tr></table>	(1, 2)	(1, 2)	(2, 2)	$\Lambda_{12} = \Phi$
(1, 2)	(1, 2)	(2, 2)			
$\phi_{13}$	<table border="1"><tr><td>(1, 2)</td><td>(2, 1)</td><td>(2, 1)</td></tr></table>	(1, 2)	(2, 1)	(2, 1)	$\Lambda_{13} = \Lambda_9$
(1, 2)	(2, 1)	(2, 1)			
$\phi_{14}$	<table border="1"><tr><td>(1, 2)</td><td>(2, 1)</td><td>(2, 2)</td></tr></table>	(1, 2)	(2, 1)	(2, 2)	$\Lambda_{14} = \Lambda_{10}$
(1, 2)	(2, 1)	(2, 2)			
$\phi_{15}$	<table border="1"><tr><td>(1, 2)</td><td>(2, 2)</td><td>(2, 1)</td></tr></table>	(1, 2)	(2, 2)	(2, 1)	$\Lambda_{15} = \Lambda_{11}$
(1, 2)	(2, 2)	(2, 1)			
$\phi_{16}$	<table border="1"><tr><td>(1, 2)</td><td>(2, 2)</td><td>(2, 2)</td></tr></table>	(1, 2)	(2, 2)	(2, 2)	$\Lambda_{16} = \Lambda_{12} = \Phi$
(1, 2)	(2, 2)	(2, 2)			

FIGURE 3b



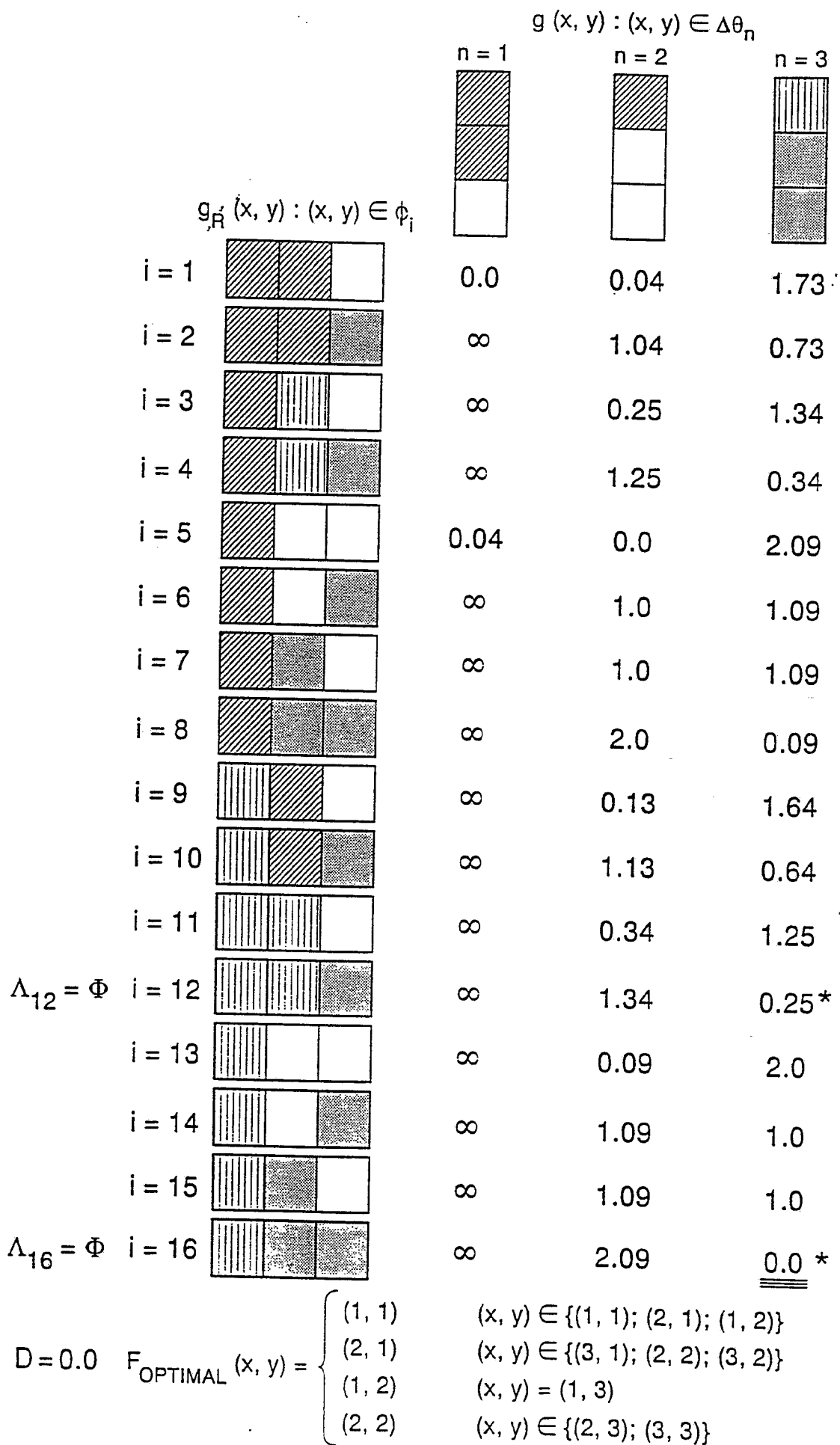





FIGURE 4

-48-

$g(x, y) : (x, y) \in \Delta\theta_n$			
	$n = 1$	$n = 2$	$n = 3$
$g_R(x, y) : (x, y) \in \phi_i$			
$\lambda_1 = \begin{cases} \phi_1 & \text{[diagonal lines, diagonal lines, white]} \\ \phi_5 & \text{[diagonal lines, white, white]} \end{cases}$	$\hat{D}_{1,1} = 0.0$	$\hat{D}_{1,2} = 0.0$	$\hat{D}_{1,3} = 1.73$
$\lambda_2 = \begin{cases} \phi_{12} & \text{[vertical lines, vertical lines, stippled]} \\ \phi_{16} & \text{[vertical lines, stippled, stippled]} \end{cases}$	$\hat{D}_{2,1} = \infty$	$\hat{D}_{2,2} = 1.73$	$\hat{D}_{2,3} = 0.0$

$$\hat{F}_{\text{OPTIMAL}}(x, y) = \begin{cases} (1, 1) & (x, y) \in \{(1, 1); (2, 1); (1, 2)\} \\ (2, 1) & (x, y) \in \{(3, 1); (2, 2); (3, 2)\} \\ (1, 2) & (x, y) = (1, 3) \\ (2, 2) & (x, y) \in \{(2, 3); (3, 3)\} \end{cases}$$

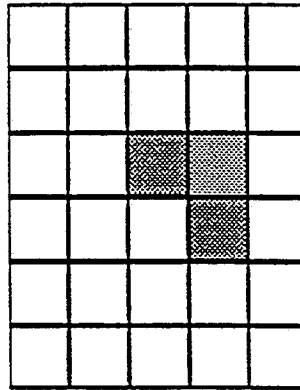
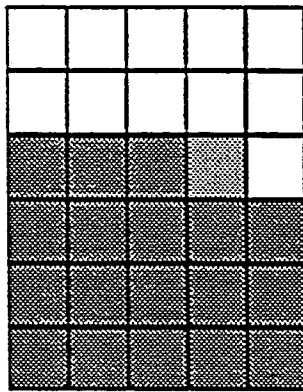
FIGURE 5

-49-

$G =$

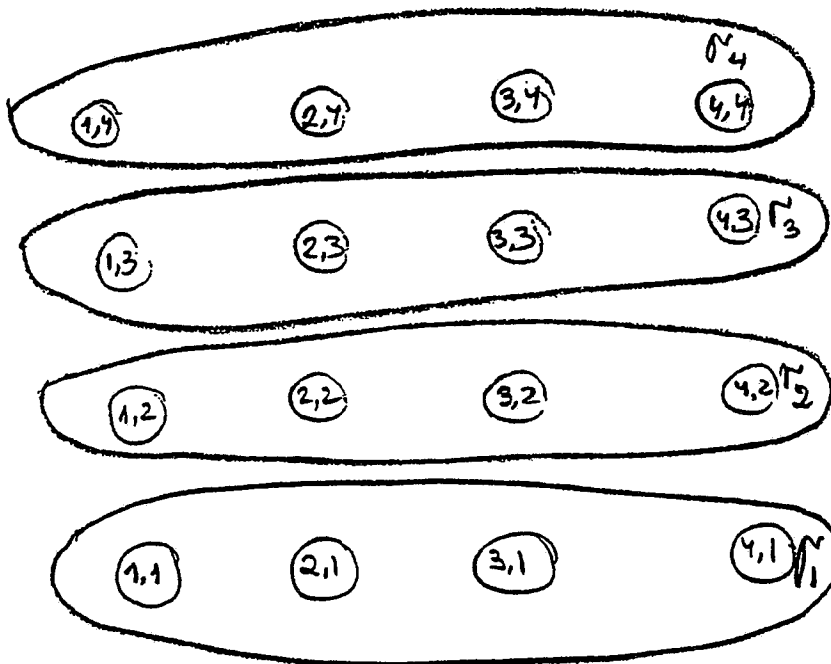
$(1, 3)$	$(2, 3)$	$(3, 3)$
$(1, 2)$	$(2, 2)$	$(3, 2)$
$(1, 1)$	$(2, 1)$	$(3, 1)$

-50-

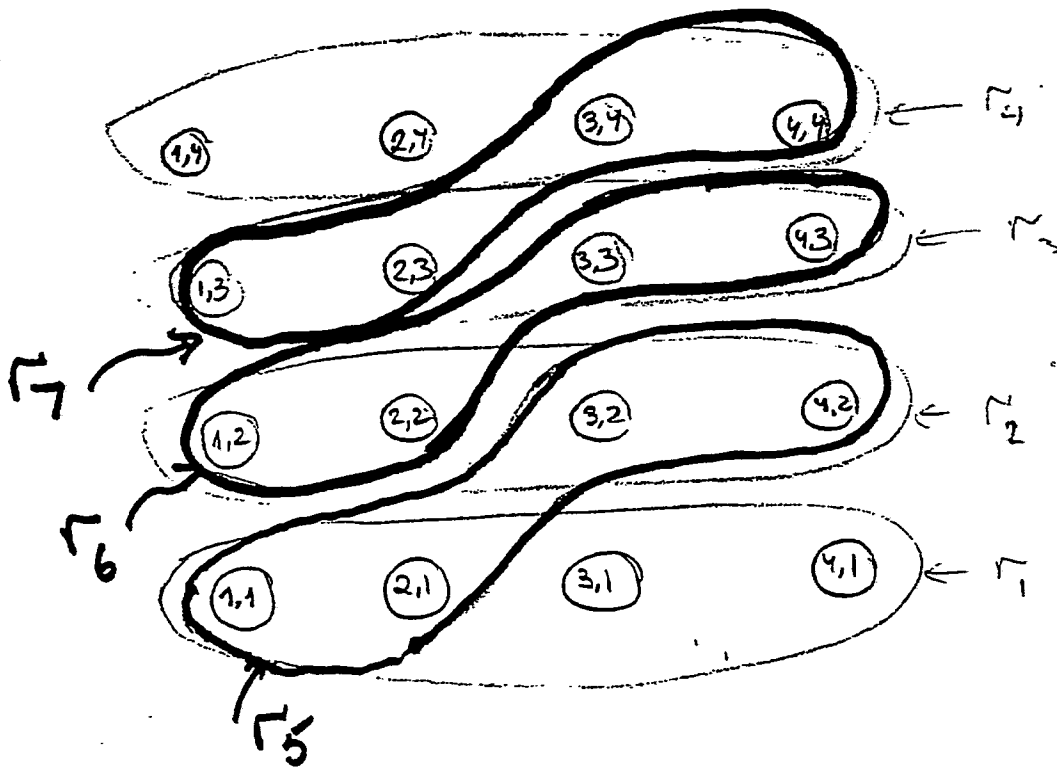


Page 7

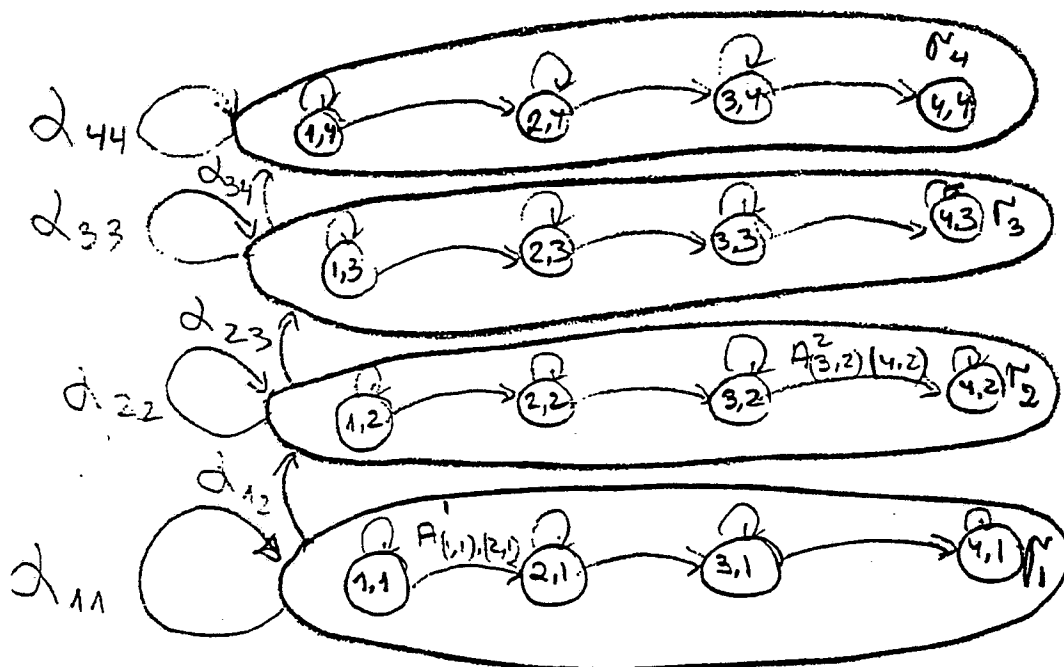
-51-



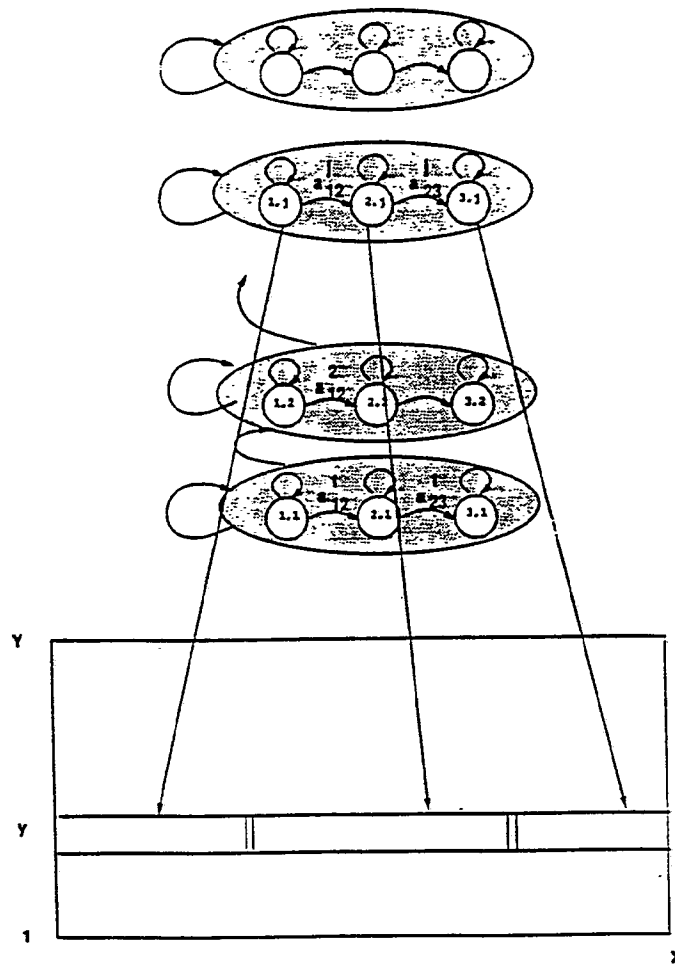
-52-



-53-



# LOCAL LIKELIHOOD CALCULATION



- Local row likelihoods  $\{l_{yj} \mid 1 \leq y \leq Y; 1 \leq j \leq N_Y\}$

$$l_{yj} = \max_{\substack{s(1:X,y) \\ s(x,y)=(\cdot,j)}} \log \Pr(g(1:X,y) \mid s(1:X,y))$$

— 1-D warping



-55-

## THE ALGORITHM

- Define

$$L_{yj} = \max_{\substack{s(1:X, 1:y) \\ s(x,y)=(\cdot, j)}} \log Pr(g(1:X, 1:y) \mid s(1:X, 1:y))$$

Global cost :  $\bullet L = L_{YN_Y}$

---

- Initialization:  $L_{1j} = l_{1j} + \log \alpha_{0j}$

```

→ • DO y = 2, Y
  → • DO j = 1, NY
    
$$L_{yj} = \max_{1 \leq m \leq N_Y} [L_{y-1,m} + \log \alpha_{mj}] + l_{yj}$$

  • ENDDO j
• ENDDO y.
```

- 1-D vertical warping with local likelihoods obtained by 1-D horizontal warpings.

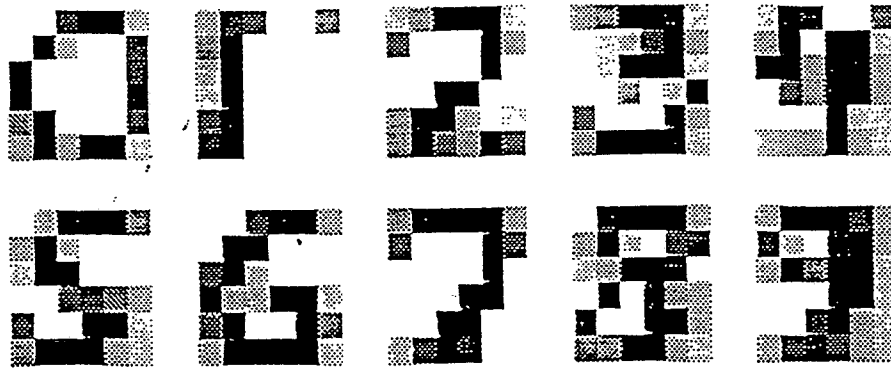
-56-

name Donnie

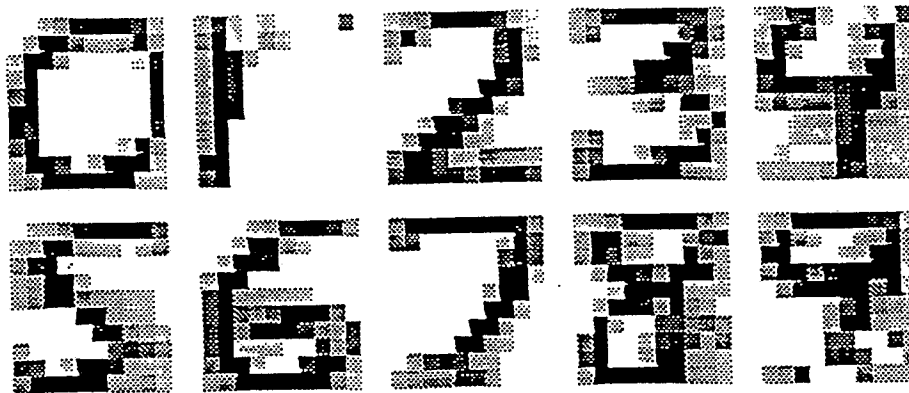
	0	1	2	3	4	5	6	7	8	9
a	0	1	2	3	4	5	6	7	8	9
b	0	1	2	3	4	5	6	7	8	9
c	0	1	2	3	4	5	6	7	8	9
d	0	1	2	3	4	5	6	7	8	9
e	0	1	2	3	4	5	6	7	8	9
f	0	1	2	3	4	5	6	7	8	9
g	0	1	2	3	4	5	6	7	8	9
h	0	1	2	3	4	5	6	7	8	9
i	0	1	2	3	4	5	6	7	8	9
j	0	1	2	3	4	5	6	7	8	9

Fig. 81

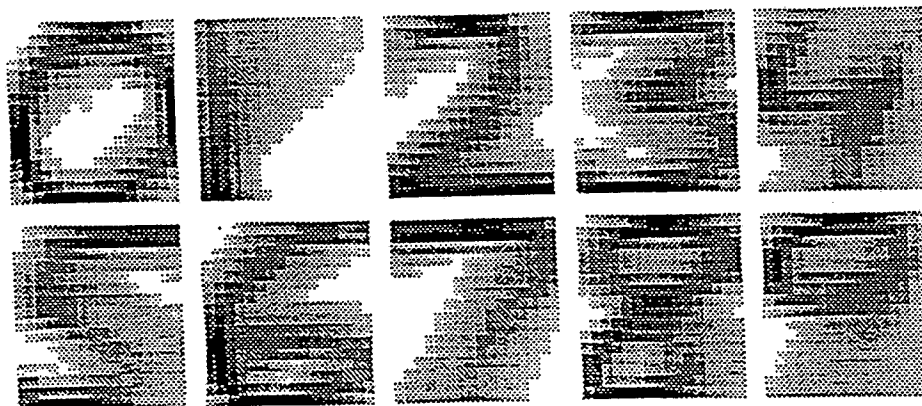
-57-



6x6



10x10



16x16

Fig. 9 12

**Claims:**

1. A method of optical character recognition, the method comprising the steps of:
  - a. storing a plurality of two-dimensional hidden Markov models, each such model comprising a one-dimensional shape-level hidden Markov model comprising one or more shape-level states, each shape-level state comprising a one-dimensional pixel-level hidden Markov model comprising one or more pixel-level states;
  - b. scanning an image to produce one or more sequences of pixels;
  - 10 c. for a stored two-dimensional hidden Markov model,
    - i. determining for each sequence of pixels a local Viterbi score for a plurality of pixel-level hidden Markov models; and
    - ii. determining a global Viterbi score of a shape-level hidden Markov model based on a plurality of local Viterbi scores and the sequences of pixels; and
  - 15 d. recognizing the scanned image based on one or more global Viterbi scores.
2. The method of claim 1 wherein the step of recognizing the scanned image comprises the step of recognizing the scanned image based on the two-dimensional hidden Markov model having the highest global Viterbi score.
- 20 3. The method of claim 1 wherein the probability of a first state in a stored two-dimensional hidden Markov model equals zero when a left neighbor state is not a member of the same pixel-level model as the first state.
4. The method of claim 1 wherein the probability of a first state in a stored two-dimensional hidden Markov model is based on the value of a left neighbor pixel-level state and the value of a bottom neighbor shape-level state.
- 25 5. An optical character recognition system, the system comprising:
  - a. a memory storing a plurality of two-dimensional hidden Markov models, each such model comprising a one-dimensional shape-level hidden Markov model

comprising one or more shape-level states, each shape-level state comprising a one-dimensional pixel-level hidden Markov model comprising one or more pixel-level states;

- b. means for scanning an image to produce one or more sequences of pixels;
- 5 c. means, coupled to the means for scanning and the memory, for determining local Viterbi scores for a sequence of pixels, each such score based on a pixel-level hidden Markov model;
- d. means, coupled to the means for determining local Viterbi scores, for determining a global Viterbi score of a shape-level hidden Markov model  
10 based on a plurality of local Viterbi scores and the sequences of pixels; and
- e. means, coupled to the means for determining a global Viterbi score, for recognizing the scanned image based on one or more global Viterbi scores.

6. The system of claim 5 wherein the means for recognizing the scanned image comprises means for recognizing the scanned image based on the two-  
15 dimensional hidden Markov model having the highest global Viterbi score.

1/3

FIG. 1

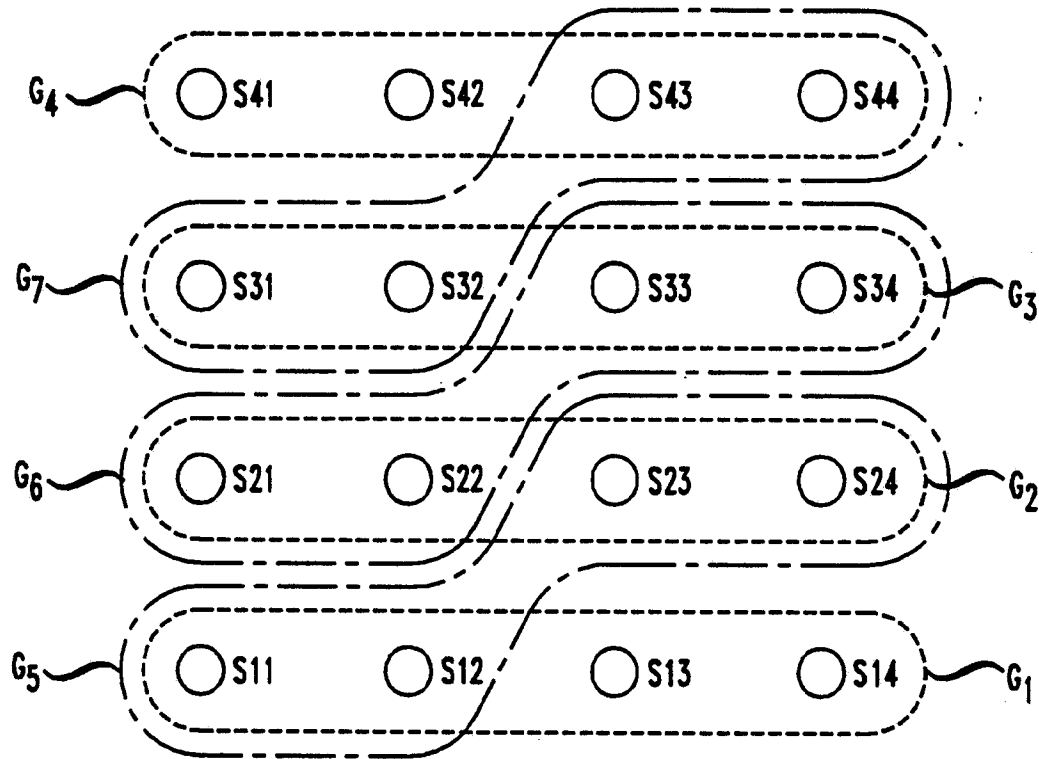
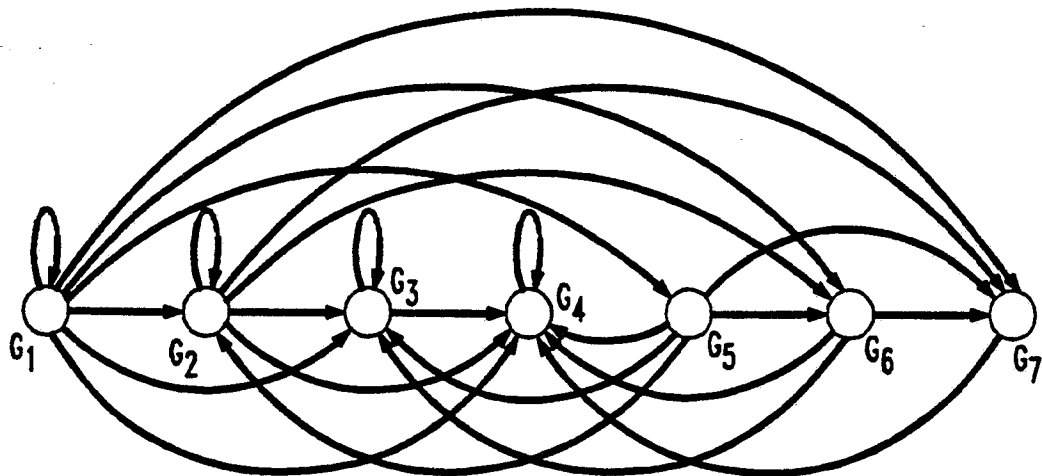
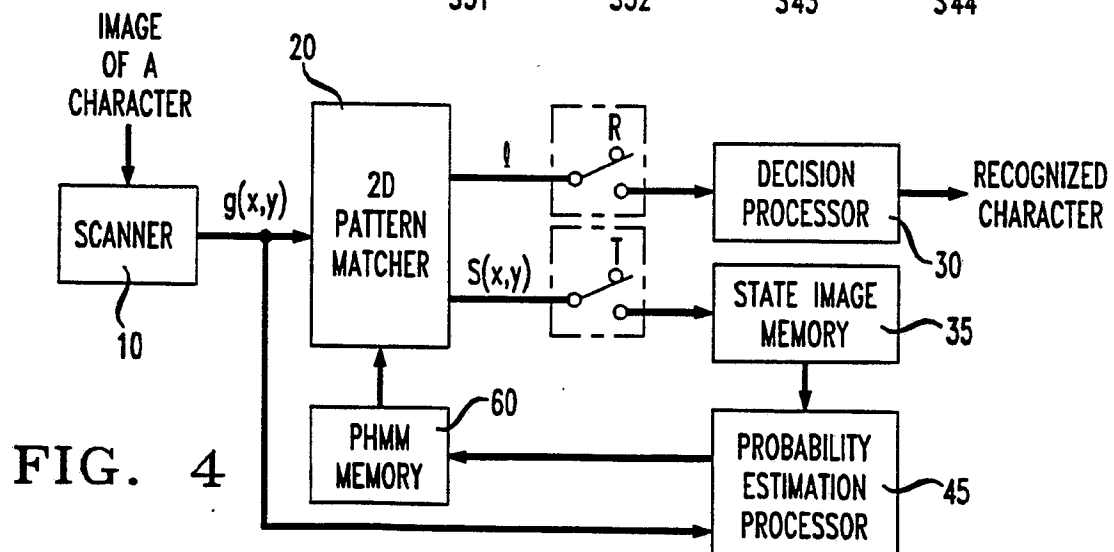
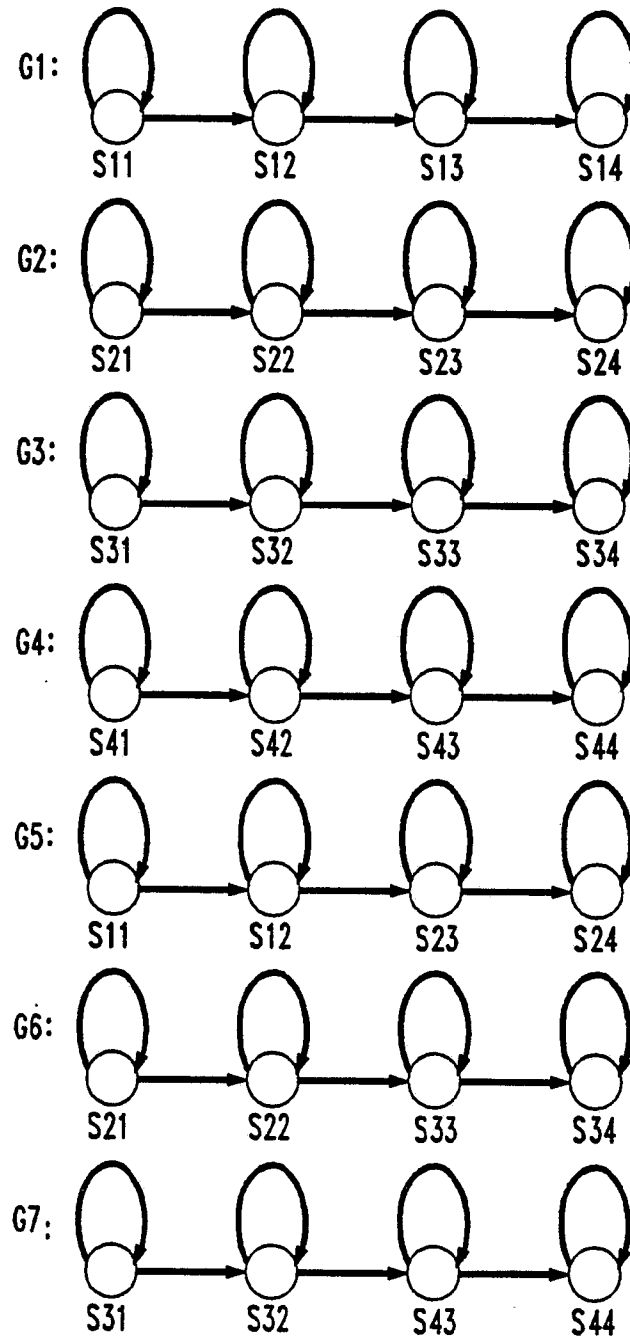


FIG. 3



2/3

FIG. 2



3/3

FIG. 5

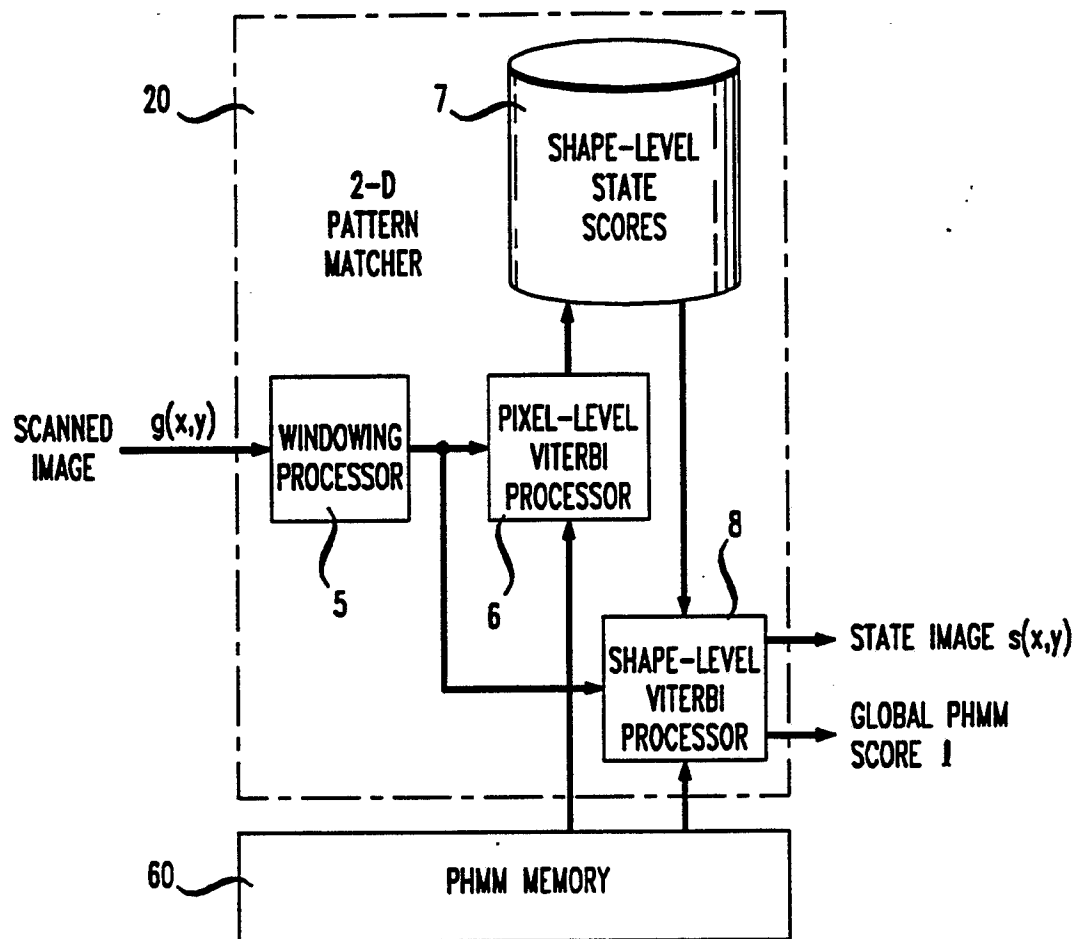
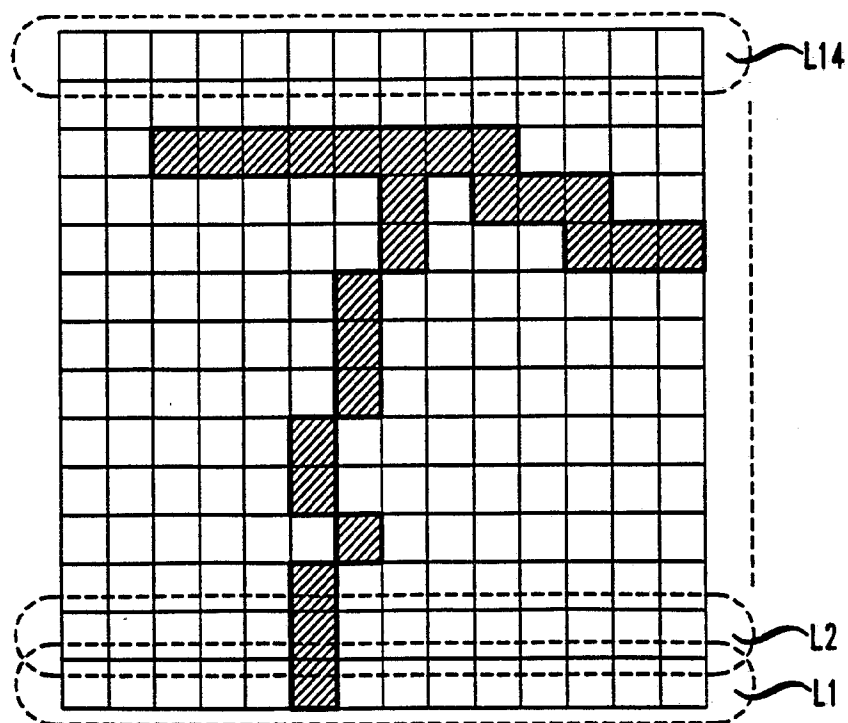


FIG. 6





**A. CLASSIFICATION OF SUBJECT MATTER**

IPC(5) :G06K 9/52

US CL :382/28

According to International Patent Classification (IPC) or to both national classification and IPC

**B. FIELDS SEARCHED**

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 382/14,15,16,28,36,37,38,39,40  
364/514,582 395/61,77

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US,A, 4,599,692 (TAN ET AL) 08 JULY 1986 See column 11, line 6 through column 12, line 68	1-6
X	US,A, 4,593,367 (SLACK ET AL) 03 JUNE 1986 See column 11, line 12 through column 12, lien 68	1-6
X	US,A, 4,599,693 (DENENBERG) 08 JULY 1986 See column 11, line 12 through column 13, line 2	1-6

☒ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:	*T* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
*A* document defining the general state of the art which is not considered to be part of particular relevance	*X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
*E* earlier document published on or after the international filing date	*Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
*L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	*Z* document member of the same patent family
*O* document referring to an oral disclosure, use, exhibition or other means	
*P* document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search

03 January 1980

Date of mailing of the international search report

13 JUL 1993

Name and mailing address of the ISA/US  
Commissioner of Patents and Trademarks  
Box PCT  
Washington, D.C. 20231

Facsimile No. NOT APPLICABLE

Authorized officer

J. L. COUSO

Telephone No. (703) 305-4774

## INTERNATIONAL SEARCH REPORT

International application No.  
PCT/US93/01843

## C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US,A, 4,620,286 (SMITH ET AL) 28 OCTOBER 1986 See column 11, line 11 through column 13, line 2	1-6